

**DESIGN AND IMPLEMENTATION OF A POWER
CONVERTER FOR SOIL VITRIFICATION APPLICATION**

by

ROBERTO CARLOS IZQUIERDO GARCIA, B.S.E.

A THESIS

IN

ELECTRICAL ENGINEERING

**Submitted to the Graduate Faculty
of Texas Tech University in
Partial Fulfillment of
the Requirements for
the Degree of**

MASTER OF SCIENCE

IN

ELECTRICAL ENGINEERING

Approved

August, 2001

ACKNOWLEDGMENT

Thanks to God for make my what I am.

Thanks to my family for their love, support and patience. To my parents who encourage me to go further, to prepare my self better. Thank to my brother and sister for encouraging me.

I want to thank Dr. Ramirez from UDLAP and Dr. Hagler from TTU for their support and help to participate in the dual program between TTU and UDLAP. Thanks to all the professors in the UDLAP who provide me with the tools to succeed in TTU.

I also want to thank Dr Dickens for his help and for the opportunity to learn from him, for him support and guidance.

Thanks to all my friends in Lubbock. You help me to grown as a person and help me to go through difficult situations.

Most of all, I want to thanks to a person who give me her love, support, patience and most of all, the strength to complete one of my most difficult challenges. She has been my reason to keep going.

Te amo, Edna Karina.

TABLE OF CONTENT

ACKNOWLEDGMENTS.....	ii
ABSTRACT.....	v
LIST OF TABLE	vi
LIST OF FIGURES	vii
CHAPTER	
1. INTRODUCTION.....	1
2. SOIL STABILIZER.....	2
2.1 Overview.....	2
3. BUCK CONVERTER.....	3
3.1 Dc-dc converter.....	3
3.2 Buck converter.....	5
3.3 Continuous conduction mode.....	6
3.4 Discontinuous conduction mode.....	8
4. SIMULATION.....	11
4.1 Buck converter.....	11
4.2 Controller.....	13
4.2.1 Voltage mode configuration.....	14
4.2.2 Current mode configuration.....	18
5. IMPLEMENTATION.....	26
5.1 Electric diagram.....	26
5.1.1 Components.....	26
5.2 Microcontroller 68HC12.....	28
5.2.1 PWM module.....	28
5.2.2 Analog to Digital converter.....	32
5.2.3 Keyboard.....	33
5.2.4 LCD.....	36
5.3 First implemented design.....	38
5.3.1 Interface Boards	40

5.3.2	Control approach.....	42
5.3.2.1	Increment-decrement..	42
5.3.2.2	Proportional control...	46
5.4	Second implemented design.....	49
5.4.1	Board.....	51
5.4.2	Control approach.....	54
5.4.3	Software.....	56
6.	CONCLUSIONS.....	60
	BIBLIOGRAPHY.....	61
	APPENDIX	
A.	68HC12 CODE.....	62
B.	PSPICE SIMULATION.....	65
C.	CONTROLLER BOARDS.....	68
D.	TORCH TRUCK.....	79

ABSTRACT

Texas Tech in conjunction with Montech Research proposed the construction of a soil stabilizer using a power graphite torch capable of generating hot plasma gas to melt the soil. The project will be used by the Texas Department of Transportation (TDOx).

The present work is a description of the electrical modules implemented to control the power of the equipment. The use of a buck converter to control the output power is proposed. Texas Tech was in charge of the construction of the electrical part of the project, a power supply capable to control 100kW.

The TDOx proposed the project because for some applications a material stronger than concrete is required, and the melted soil is three times stronger than concrete.

The results observed on the field shows that it is possible to melt down the soil with an output of 80kW. With further improvement on the design of the power modules it is possible to control up to 100kW.

LIST OF TABLES

4.1	Simulation cases.....	21
5.1	Input keyboard for the matrix board.	35
5.2	LCD Instruction set.....	38
5.3	68HC12 ports.....	52
5.4	Multiplexer outputs.	54
5.5	Keys function.....	58

LIST OF FIGURES

2.1	Soil stabilizer overview.....	2
3.1	Switch mode dc-dc conversion.....	3
3.2	Pulse width modulator: (a) block diagram; (b) comparator signals...	4
3.3	Step-down dc-dc converter.....	5
3.4	Step-down converter states: (a) switch on: (b) switch off.....	6
3.5	Continuous conduction mode.....	6
3.6	Boundary between the continuous and discontinuous mode.....	8
3.7	Discontinuous conduction in step-down converter.....	8
3.8	Output voltage ripple in a step-down converter.	10
4.1	Buck converter model for PSpice.....	11
4.2	Simulation of the model at $D = 50\%$	12
4.3	Discontinuous conduction mode.	13
4.4	Voltage control mode.	14
4.5	Voltage controller mode.....	14
4.6	Plasma burner.....	15
4.7	Current control with one module.	15
4.8	Error signal.....	16
4.9	Current control with three modules: $I_{set} = 200A$	17
4.10	Current control with different setpoints.....	17
4.11	Current control: (a) Controller, (b) PWM waveform.....	18
4.12	Use of the flip-flop as the controller.....	19
4.13	PWM control signals: (a) start, (b) reset and (c) clock signal.....	19
4.14	Current mode.....	20
4.15	Setpoint of 200A.....	20
4.16	Equal setpoints and inductances.....	21
4.17	Different setpoints, equal inductances.....	22
4.18	Low load inductance.....	22
4.19	Different setpoint, low load inductance.....	23

4.20	Same setpoints, different inductances.....	23
4.21	Same setpoint, different inductances.....	24
4.22	Equal inductances, same setpoint.....	24
4.23	Equal inductances, same setpoint, low load inductance.....	25
5.1	Electric diagram.....	26
5.2	Rectifier bridge (a) symbolic and (b) internal diagram.....	27
5.3	IGBT SkiiP 1442. Buck converter.....	27
5.4	Left-aligned waveform, positive polarity.	29
5.5	PWM clock chains.....	30
5.6	PWM left-aligned output channel.	31
5.7	Analog-to-Digital Converter.	32
5.8	4 x 4 Matrix keyboard.	33
5.9	Schematic keyboard connections.....	34
5.10	LCD Connections.....	37
5.11	Address lines on the LCD.....	37
5.12	Block diagram.....	39
5.13	First design of electrical diagram.....	39
5.14	Interface board with the HC12.....	40
5.15	Configuration for the optocouplers.	40
5.16	Lowpass filter.....	41
5.17	Filter design using Simplorer: (a) model; (b) simulation.....	42
5.18	Increment-decrement.....	43
5.19	Buck converter on Simplorer.....	44
5.20	Control stages on Simplorer.....	44
5.21	Simulation with no initial current on the inductance.....	45
5.22	Initial condition, $I_L = 400A$	46
5.23	Feedback diagram.....	46
5.24	Sampling model.....	47
5.25	Current and Duty cycle.....	48
5.26	Second design: (a) Main circuit; (b) interior of the module.....	50

5.27	Second interface.....	51
5.28	Internal diagram of the controller board.....	52
5.29	Controller module interface.....	53
5.30	New PWM controller.....	55
5.31	Second configuration.....	55
5.32	Software diagram.....	56
5.33	Display shell.....	57
5.34	Registers relations.	58
5.35	Display view: (a) Power mode and (b) Current mode.....	58
B.1	Plasma burner.....	66
B.2	Current mode controller.....	67
C.1	68HC12 interface with the IGBTs.	69
C.2	Pcboard silk side of diagram on Figure C.1.....	70
C.3	Pcboard component side of diagram on Figure C.1.....	70
C.4	Pcboard solder side of diagram on Figure C.1.....	70
C.5	Schematic of the analog to digital interface.....	71
C.6	Pcboard silk side of diagram on Figure C.5.....	72
C.7	Pcboard component side of diagram on Figure C.5.....	72
C.8	Pcboard solder side of diagram on Figure C.5.....	72
C.9	Controller box interface between the HC12 and the keyboard-LCD.	73
C.10	Pcboard silk side of diagram on Figure C.9.....	74
C.11	Pcboard component side of diagram on Figure C.9.....	74
C.12	Pcboard solder side of diagram on Figure C.9.....	74
C.13	PWM controller schematic.....	75
C.14	Pcboard silk side of diagram on Figure C.13.....	76
C.15	Pcboard component side of diagram on Figure C.13.....	76
C.16	Pcboard solder side of diagram on Figure C.13.....	76
C.17	Controller interface.....	77
C.18	Analog interface.....	77
C.19	68HC12 General interface.....	78

C.20 Box cover..... 78

D.1 Mechanical system..... 80

D.2 Transformer..... 80

D.3 Power converter operation..... 81

CHAPTER 1

INTRODUCTION

A relatively new branch of electronics is power electronics, where the control of hundred of volts and amperes is achieved with new semiconductor devices such as power diodes, BJT, Power MOSFETS, thyristors and Insulated Gate Bipolar Transistors (IGBT) [1]. Such devices allow us to achieve a better performance on diverse applications and also allow new control techniques based on this technology. One branch is related to the design of power supplies.

A power supply is a device capable of maintaining a constant voltage or current delivered to a load using diverse techniques. Among the power supplies, there are the dc-dc topologies, which have a constant voltage over a period of time. The control of the output power becomes a problem when the load demands high power from the input, then risking the integrity of the application. Therefore it is important to have an adequate control over the load power to ensure the integrity of the application.

To convert and control the voltage we have different topologies. Among them there is one topology known as the Buck Converter, also known as a Step Down Converter. The buck converter is a good option to control a dc-dc voltage, because it is simple and efficient. The buck converter is a voltage controller whose voltage output is lesser than or equal to the input voltage.

CHAPTER 2

SOIL STABILIZER

2.1 Overview

The SOVIT system consists in a field-deployable apparatus for sub-grade repair and stabilization by vitrifying soil in place. The operation of the systems starts with the deployment of the apparatus at the selected position. The torch consists of two concentric graphite electrodes that are placed at the ground. Then, the shroud is placed around the torch and is connected to the exhaust system. All the pumps are activated and the power is applied to the torch. The melt process is accomplished by striking an arc between the electrodes to create hot plasma-gas. The gas is forced to the tip of the torch by injecting air through the head. It is possible to inject additives to vary the melt characteristics. Due to the high temperature reached on the tip, the head assembly is cooled down by an internal water supply system. The off gas generated by the process is collected through the shroud and routed to the spray quencher and scrubber system for treatment. Figure 2.1 shows the block diagram of the main components of the Soil Stabilizer.

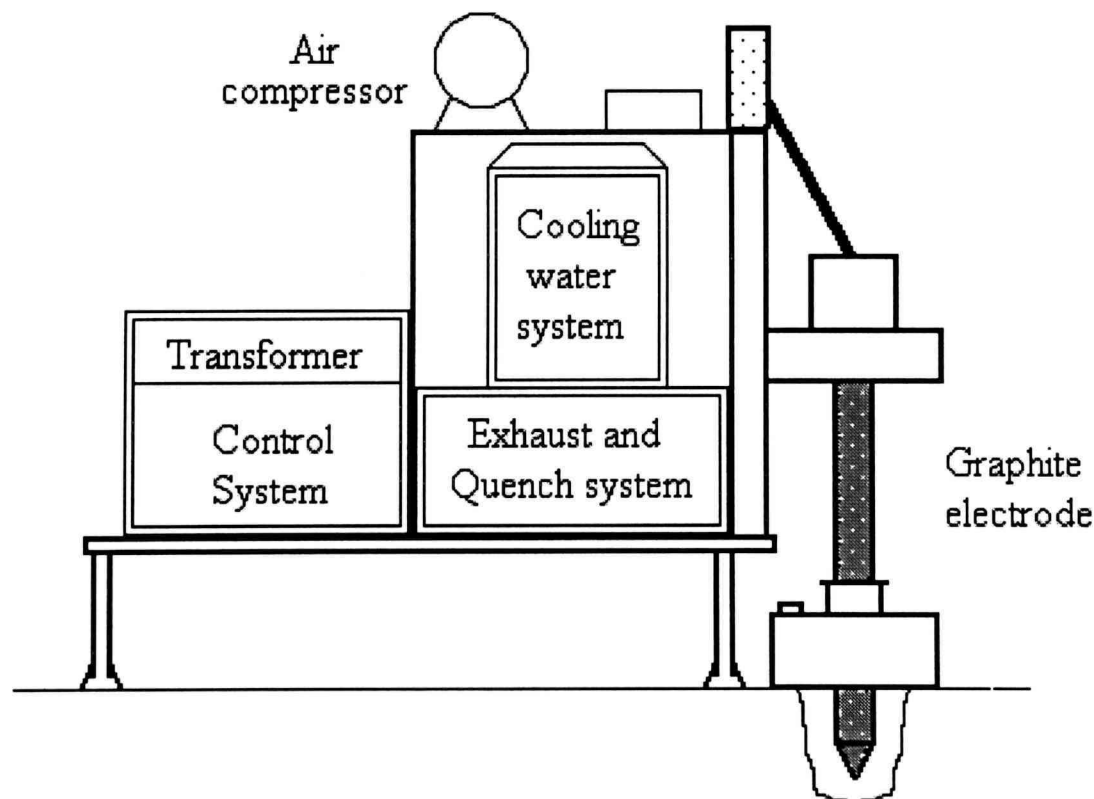


Figure 2. 1 Soil stabilizer overview.

CHAPTER 3

BUCK CONVERTER

3.1 Dc-dc Converter

Nowadays, the dc-dc is widely used in regulated power supplies and in dc motors drives. In most applications, the voltage input of these converters comes from non-regulated dc sources. Switch mode dc-dc converters are used to control the output voltage, converting them from unregulated to regulated dc input at a desired voltage level. Some dc-dc converters are:

- a. Step-down (buck) converter,
- b. Step-up (boost) converter,
- c. Step-down/step-up converter,
- d. Cuk converter.

The first two converters are basic configurations while the third and fourth are a combination of the first two.

On dc-dc converters, the average output voltage must be controlled to obtain a desired level, even if the input voltage fluctuates. A switch mode dc-dc converter uses one or more switches to convert a dc level to another. In a dc-dc converter with a defined input voltage, the average output voltage is controlled by controlling the switch on and off duration (t_{on} and t_{off}). Lets consider a basic dc-dc converter as shown in Figure 3.1. The average output voltage V_o depends on t_{on} and t_{off} . One method to control the output voltage uses a constant switching frequency and therefore a constant time period ($T_s = t_{on} + t_{off}$). In this method, called *pulsed-width modulation* (PWM) switching, the relation between the on duration over the switching time period D , is varied [1].

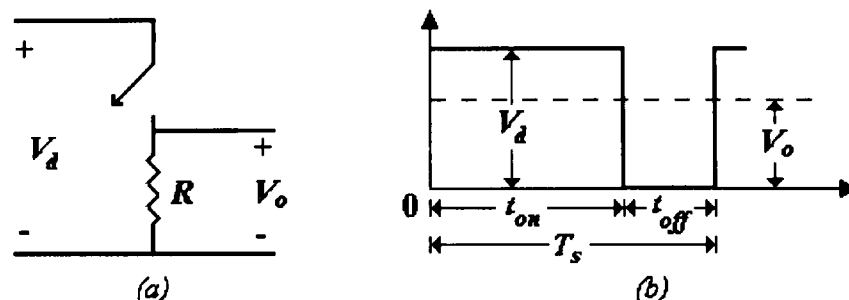


Figure 3.1 Switch mode dc-dc conversion.

In the PWM switching at a constant frequency, the switch control signal that controls the state of the switch (on or off) is produced by comparing a signal level voltage control $v_{control}$ and a frequency constant repetitive waveform, as shown in Figure 3.2b.

The control signal is generated by the amplification of the difference between the desired voltage and the actual output voltage. The frequency of the repetitive waveform determines the switching frequency. This signal can go from the hundreds to the thousand of hertz. When the control signal, which varies slower than the sawtooth is greater than the sawtooth waveform, the controls signal turns high. Thus causing the switch to turn on. Otherwise, the switch is off. In terms of $v_{control}$ and the sawtooth waveform peak, the duty cycle can be expressed as

$$D = \frac{t_{on}}{T_s}$$

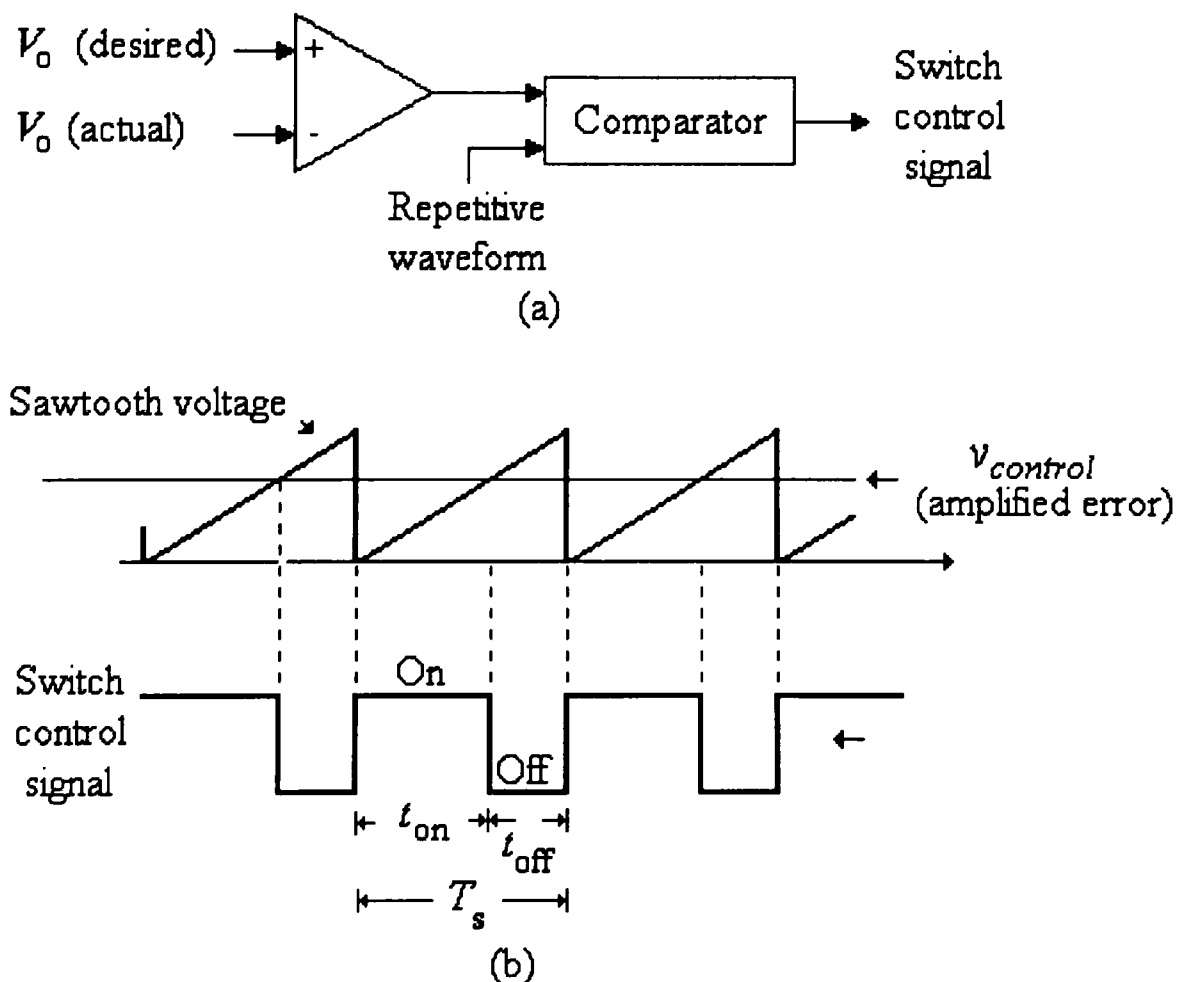


Figure 3.2 Pulse width modulator: (a) block diagram; (b) comparator signals.

3.2 Step-down (Buck) converter

A step-down converter produces a lower average output voltage than the dc input voltage. Its main application is in regulated dc power supply and dc motor speed control [1]. Based on the diagram of Figure 3.1a, the basic operation of the step-down converter can be explained. Assuming an ideal switch and a purely resistive load, the instantaneous output voltage will be as shown on Figure 3.1b, where the output voltage depends on the switch position. The average output voltage can be calculated in terms of the switch duty ratio:

$$V_o = \frac{1}{T_s} \int_0^{T_s} v_o(t) dt = \frac{1}{T_s} \left(\int_0^{t_{on}} V_d dt + \int_{t_{on}}^{T_s} 0 dt \right) = \frac{1}{T_s} \left(V_d \int_0^{t_{on}} dt \right) = \frac{t_{on}}{T_s} V_d = D V_d.$$

By varying the duty ratio t_{on} / T_s of the switch, V_o can be controlled. The circuit of Figure 3.1a has two inconvenient on real applications, (1) the load will be inductive, and (2) the output voltage fluctuates between 0 and V_d , something not possible on some applications. Even with a resistive load, we can assume to have an inductance associated with the circuit. This energy would have to be absorbed (or dissipated) by the switch, and therefore it can be destroyed.

To overcome these problems, we can use a diode and a low-pass filter (an inductor and a capacitor) as shown in Figure 3.3. During the interval when the switch is on, the diode become reverse biased and the input voltage provides energy to the load and the inductor. When the switch is off, the diode becomes forward biased and the current of the inductor flows through the load, transferring some energy to it.

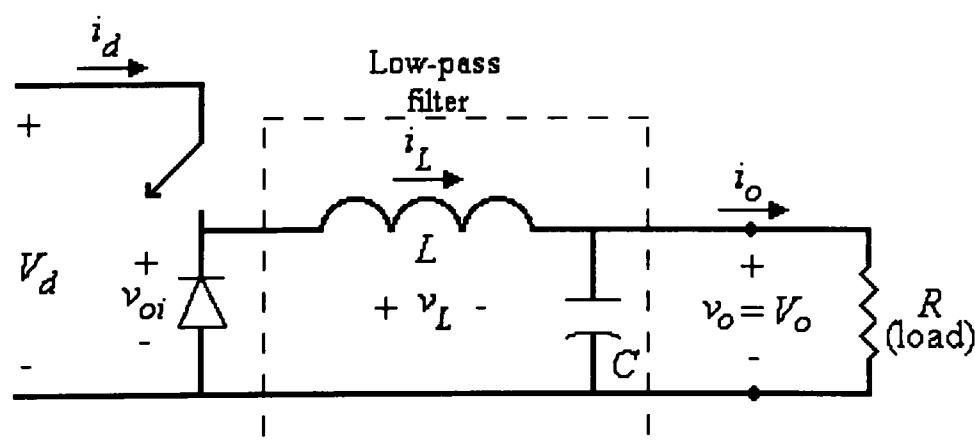


Figure 3.3 Step-down dc-dc converter.

The capacitor in the filter has to be large for applications where the instantaneous voltage at the output needs to be almost constant, $v_o \cong V_o$. From Figure 3.3, we can observe that the average inductor current i_L is equal to the average output current I_o since the average capacitor current in steady state is zero [1].

3.3 Continuous conduction mode

For the circuit of the Figure 3.3, when the switch is on for a time duration t_{on} , the switch conducts the inductor current and the diode become reverse biased. This results in a positive voltage $v_L = V_d - V_o$ across the inductor. This voltage causes a linear increase in the inductor current i_L . When the switch is closed, the diode become forward biased, and the inductive energy stored on the inductor, i_L , now flows through the diode, and $v_L = -V_o$. Figure 3.4 shows the two states of the buck converter, when the switch is on (Fig. 3.4a) and when it is off (Fig. 3.4b).

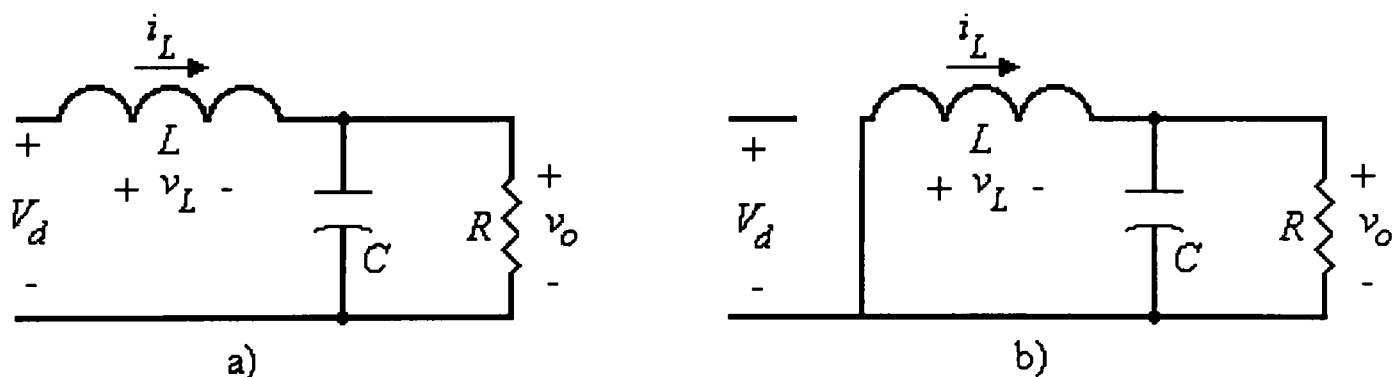


Figure 3.4 Step-down converter states: (a) switch on: (b) switch off.

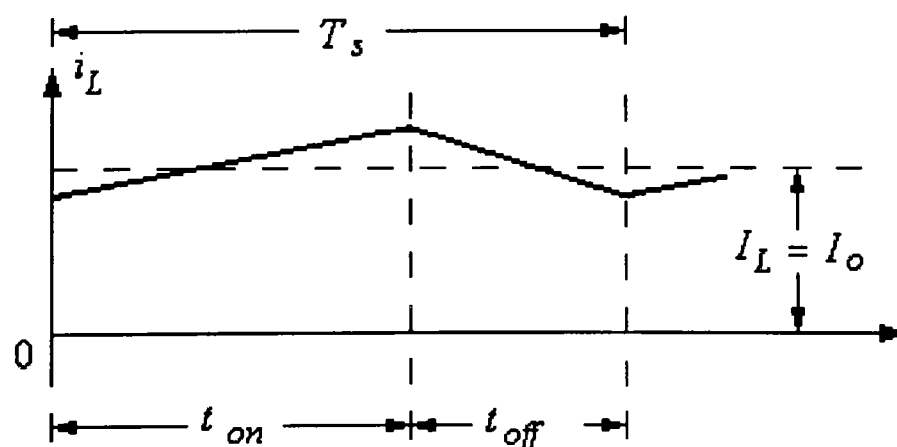


Figure 3.5 Continuous conduction mode.

The integral of the inductor voltage over a period of time must be zero.

$$\int_0^{T_s} v_L dt = \int_0^{t_{on}} v_L dt + \int_{t_{on}}^{T_s} v_L dt = 0$$

Therefore:

$$(V_d - V_o)t_{on} = V_o(T_s - t_{on})$$

$$V_d t_{on} - V_o t_{on} = V_o T_s - V_o t_{on}$$

$$V_d t_{on} = V_o T_s$$

or

$$\frac{V_o}{V_d} = \frac{t_{on}}{T_s} = D.$$

In this mode, the output voltage varies linearly with the duty cycle of the switch. Neglecting power losses associated to the circuit, the input power P_d equals the output power P_o :

$$P_d = P_o$$

$$V_d I_d = V_o I_o.$$

In continuous conduction mode, the step-down converter behaves as a transformer where the turns ratio of the equivalent transformer can be controlled electronically in a range from 0 to 1 by controlling the duty cycle. When the current of the inductor goes to zero at the end of the t_{off} period, as shown in Figure 3.6, the converter is at the boundary of the continuous mode. At this boundary, the average inductor current, where the subscript B refers to the boundary, is

$$I_{LB} = \frac{1}{2} i_{L,peak} = \frac{t_{on}}{2L} (V_d - V_o) = \frac{DT_s}{2L} (V_d - V_o) = I_{oB}.$$

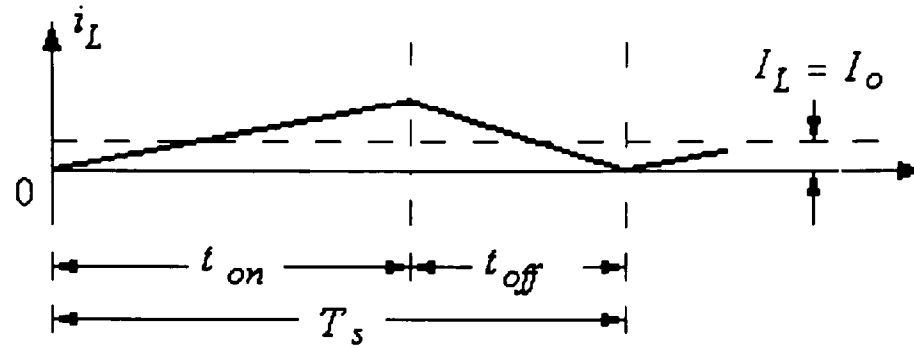


Figure 3.6 Boundary between the continuous and discontinuous mode.

Therefore, if the average output current becomes less than I_{LB} , then i_L becomes discontinuous.

3.4 Discontinuous conduction mode

In this mode, either the input voltage V_d or the output voltage V_o remains constant during the operation. Figure 3.7 shows the discontinuous mode for both cases. Only the case with constant input voltage will be discussed.

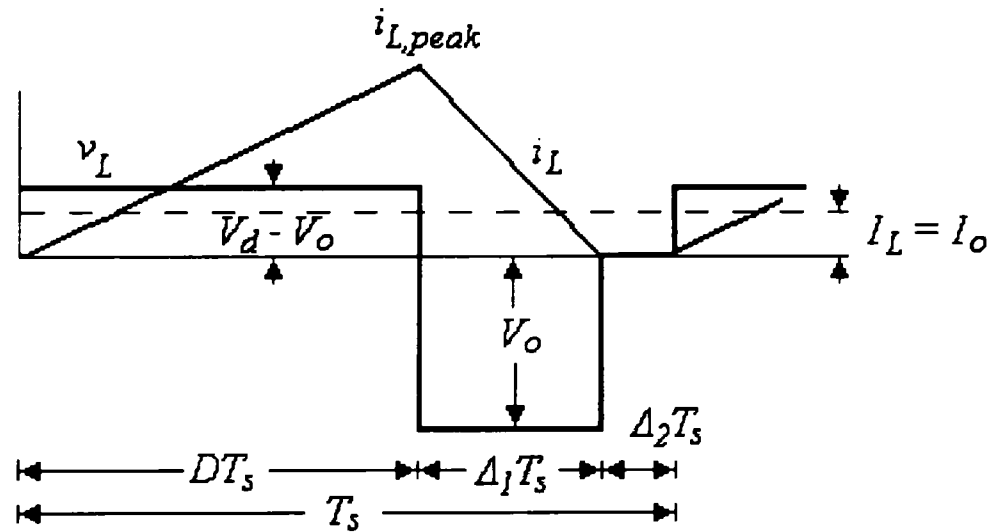


Figure 3.7 Discontinuous conduction in step-down converter.

In applications where the input voltage V_d is constant, the output voltage V_o is controlled by adjusting the value of D . From last equation and because $V_o = DV_d$, the average inductor current at the edge of the continuous-conduction mode is

$$I_{LB} = \frac{T_s V_d}{2L} D(1 - D).$$

The maximum output current that is required to be on the boundary of the continuous mode is found at $D=0.5$ (assuming that we maintain constant the parameters values). At $D=0.5$, the maximum output current required is

$$I_{LB,\max} = \frac{T_s V_d}{8L}.$$

The voltage ratio can be calculated in the discontinuous mode. Lets assume that the converter is operating at the boundary of the continuous mode, as in Figure 3.6, for given values of T , L , V_d . If these parameters are left constant and the output load power decreases (i.e. the load resistance goes up), then the average inductor current will decrease, and therefore the output voltage will increase, as shown in Figure 3.7. This results in a discontinuous inductor current. During the interval $\Delta_2 T_s$ where the inductor current is zero, the power to the load resistance is supplied by the filter capacitor. The inductor voltage at this interval is zero.

3.5 Output voltage ripple

For the previous analysis, the output capacitor is assumed to be large enough as to achieve $v_o(t) = V_o$. With a continuous conduction mode as Figure 3.8 shows, the output volt ripple can be calculated. The peak to peak voltage ripple can be written as

$$\Delta V_o = \frac{1}{C} \frac{1}{2} \frac{\Delta I_L}{2} \frac{T_s}{2}$$

where

$$\Delta I_L = \frac{V_o}{L} (1 - D) T_s.$$

By substituting ΔI_L on ΔV_o gives

$$\Delta V_o = \frac{T_s}{8C} \frac{V_o}{L} (1-D) T_s$$

where the switching frequency $f_s = 1/T_s$ and

$$f_s = \frac{1}{2\pi\sqrt{LC}}.$$

The previous equations shows that voltage ripple can be minimized by selecting a corner frequency f_c of the low pass filter at the output such that $f_c \ll f_s$. The output is independent of the output load power as long as the converter operates in the continuous conduction mode.

It has to be noted that the ripple of current on the inductor is independent of the capacitor at the output. The factors that determine the inductances ripple current are the value of the inductance, the duty cycle and the switching frequency. If we assume that the switching frequency is constant, the inductance will determine most of the current ripple.

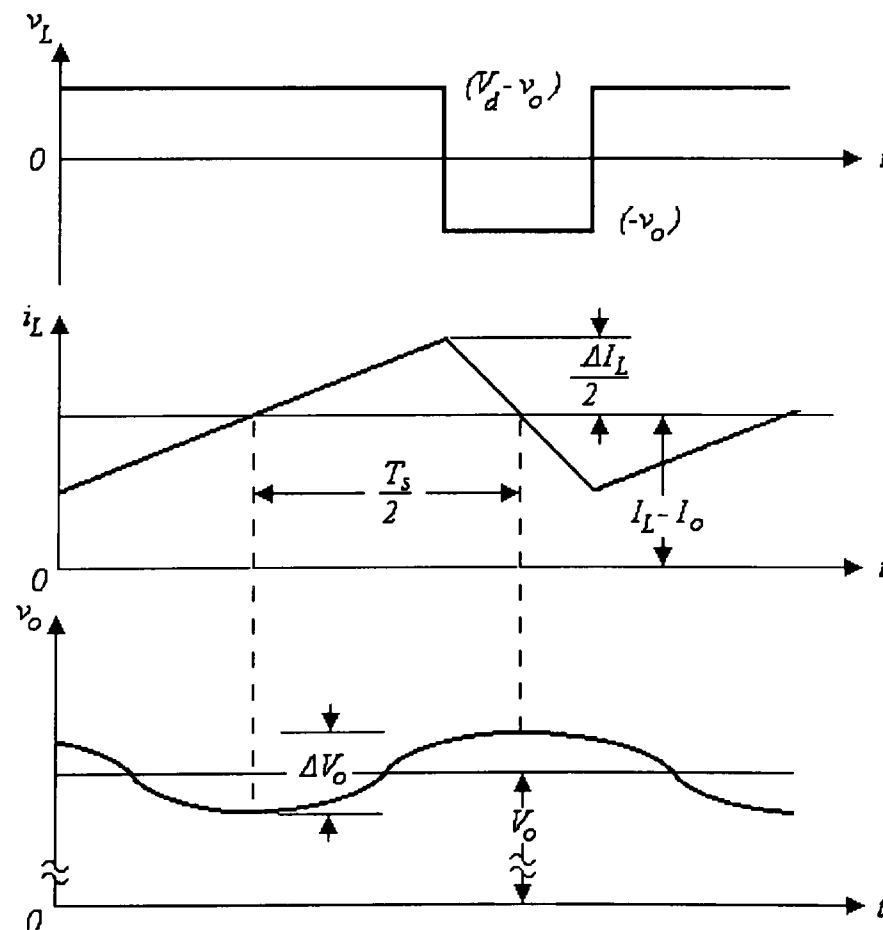


Figure 3.8 Output voltage ripple in a step-down converter.

CHAPTER 4

SIMULATION

The present chapter will present simulations of a buck converter and for the control to be implemented on the power converter. The details of the power converter will be explained on chapter 5. The models presented here are based on libraries from the course Power Electronics imparted by Dr Michel Giesselman

4.1 Buck converter

The buck converter features were explained on section 3.2. The basic components of the buck converter are presented on Figure 4.1.

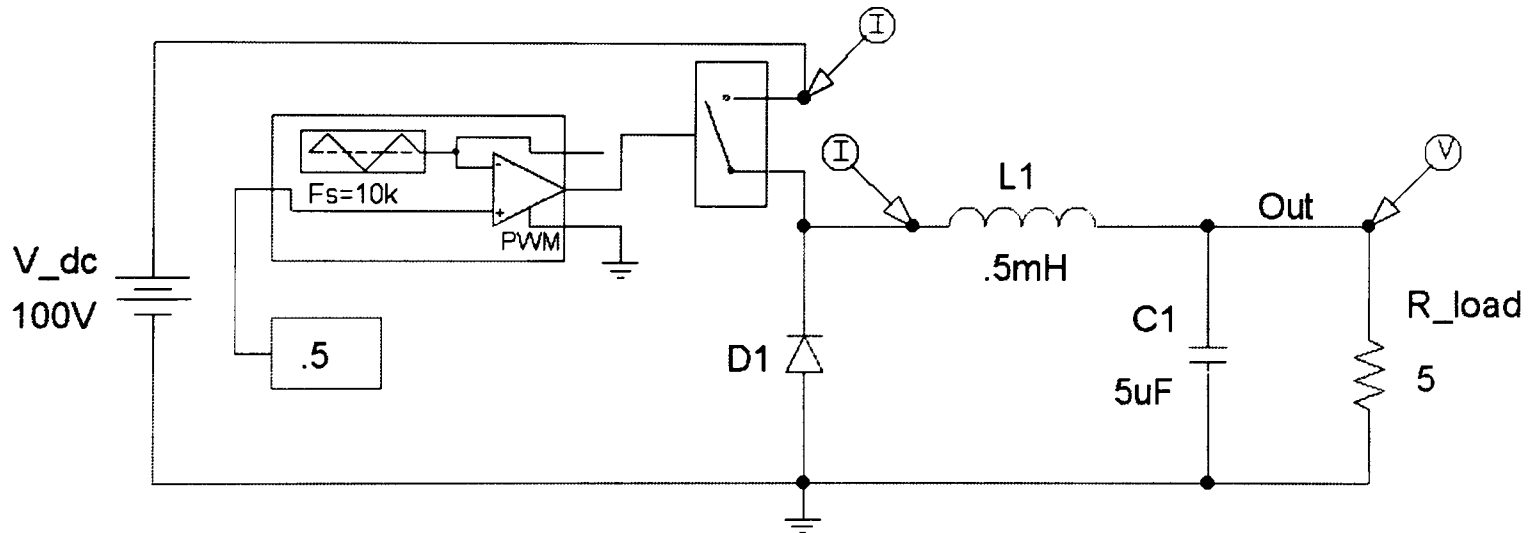


Figure 4.1 Buck converter model for PSpice

The previous model presents a switch with a variable resistance. The resistance is high if the switch is open and it is low if it is closed. This model has a PWM generator that depends on a constant as a setpoint. The PWM generator has an internal triangular signal with variable frequency and an amplitude from 0 to 1. Therefore, the setpoint for the model is a constant that varies from 0 to 1. The setpoint determines the output voltage. According to

$$V_o = D_{ty} * V_{in}$$

The buck converter is modeled with a load resistance of 5Ω , an inductance of $.5\text{mH}$ and a capacitance of $5\mu\text{F}$. For these values, we can calculate the maximum current through the inductance (at a duty cycle of 50%).

$$\Delta I_{L,\max} = \frac{T_s V_d}{8L} = \frac{(100\mu\text{s})(100\text{V})}{8(.5\text{mH})} = 5\text{A}$$

The results of the simulation for the load are presented on the Figure 4.2.

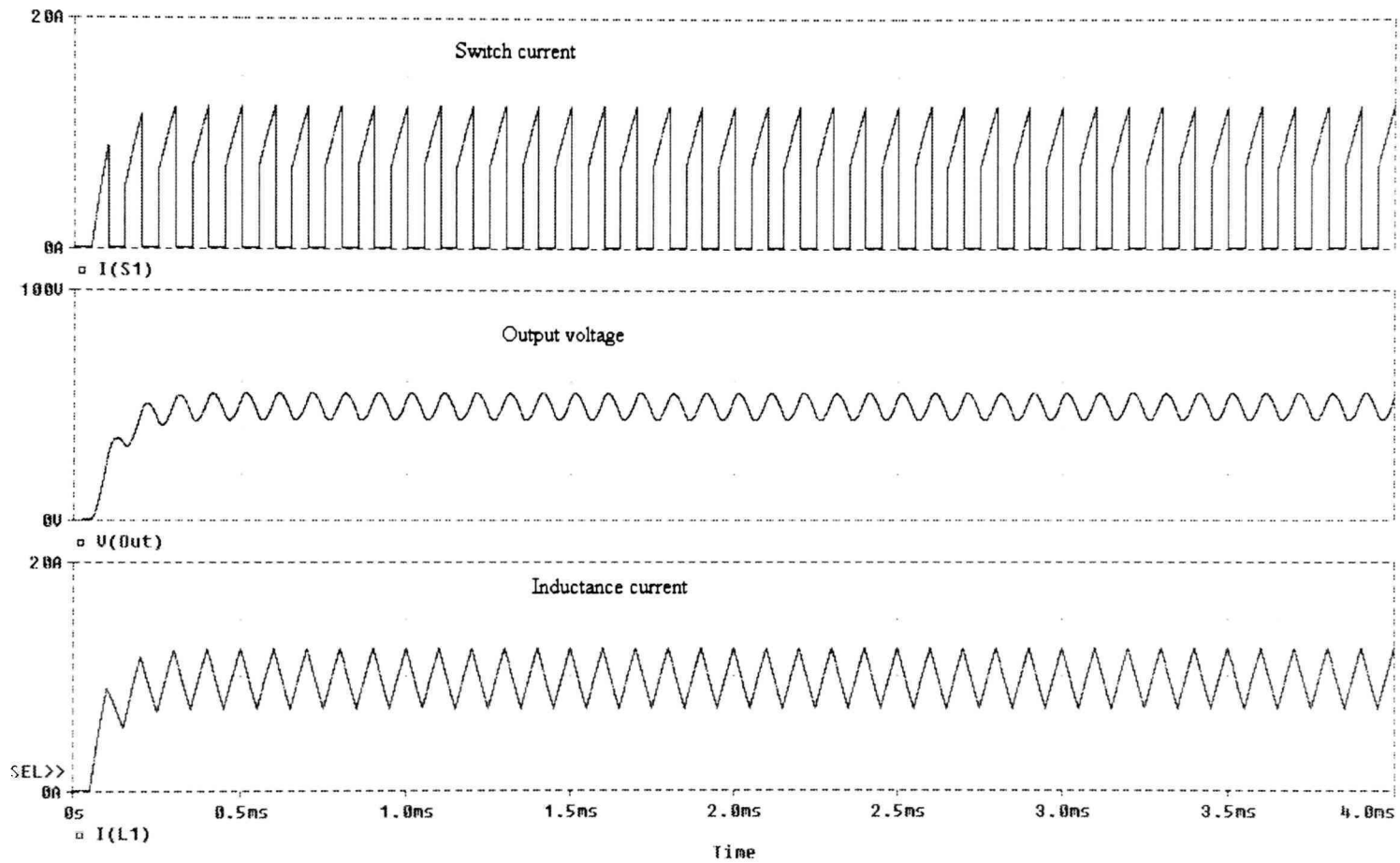


Figure 4.2. Simulation of the model at $D = 50\%$.

From the previous simulation (at $D=50\%$) we can observe that the output voltage is as desired, 50 volts, and that the current ripple on the inductor is as calculated, 5 amps. The current on the load is 10A. The minimum current on the load to maintain the buck converter on continuous mode is

$$\frac{\Delta I_L}{2} = 2.5\text{A}.$$

If we go below this point, the output voltage will not follow the duty cycle (see Figure 4.3), where the load is larger and therefore the current is below the minimum current to maintain the continuous conduction mode.

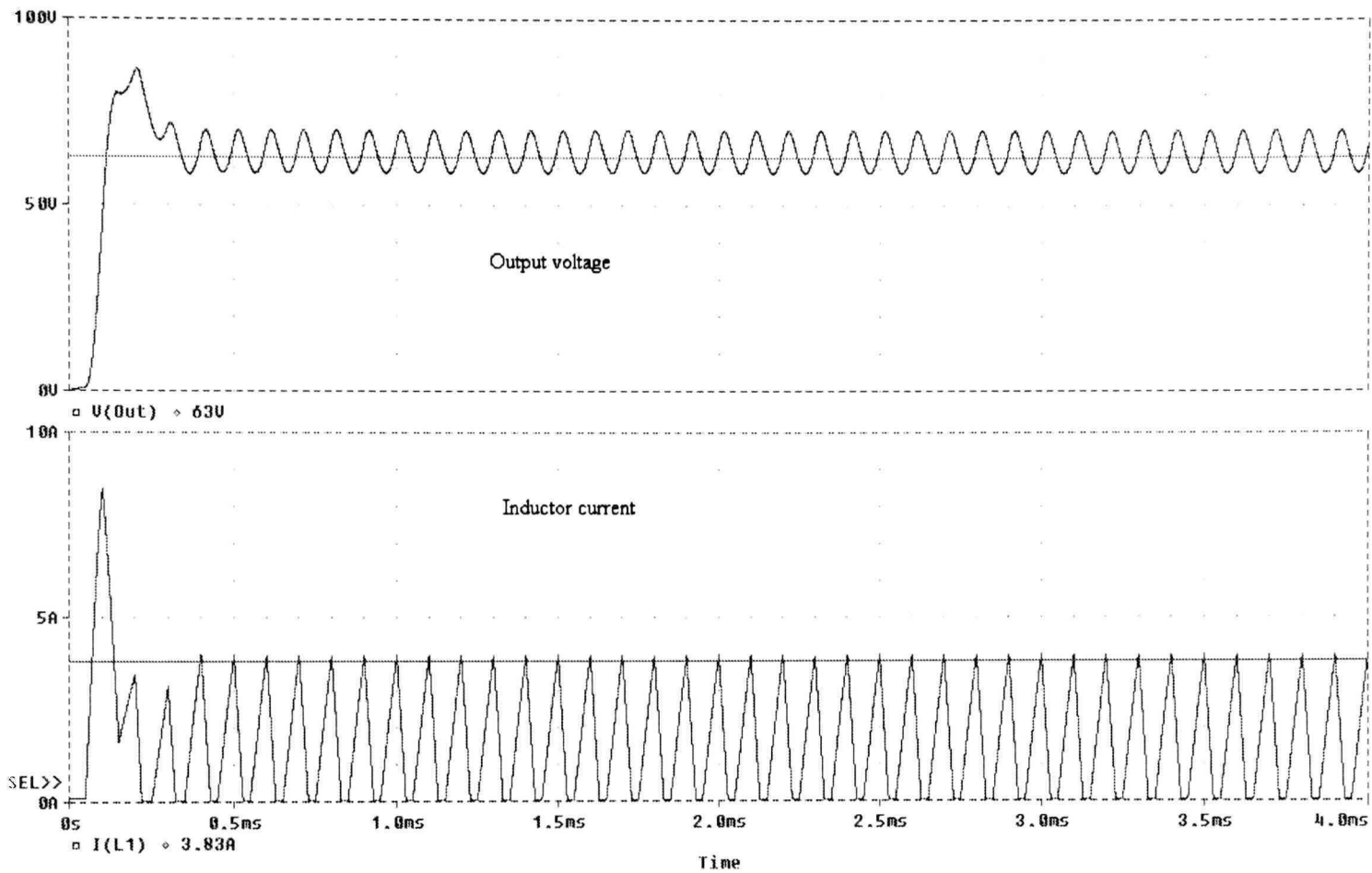


Figure 4.3 Discontinuous conduction mode.

Even though the voltage ripple is determined by the capacitor, it is the inductor what determines the conduction mode for the load.

4.2 Controller.

The next simulations are done based on the Second Design implemented, which will be explained with more detail in section 5.4. The controller module is based on an integrated circuit, the UC1823A from Unitrode Products. Although the controller can be set on two modes (voltage and current control) the variable to control is the current of the power converter. For this purpose, a current sensor is used.

4.2.1 Voltage mode configuration.

As previously said, although the configuration on the chip is set as voltage mode, it was the current the variable to control. Figure 4.4 shows the configuration for the control.

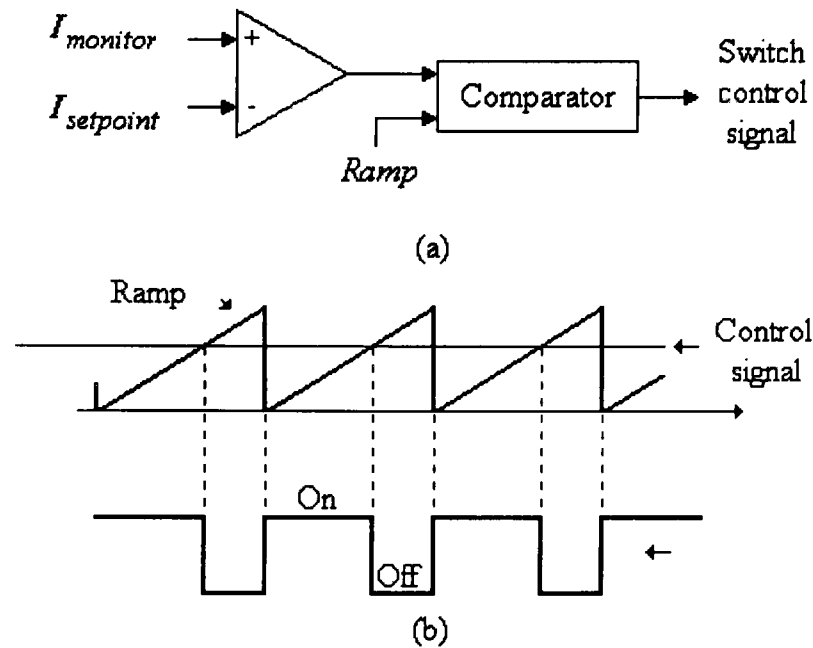


Figure 4.4 Voltage control mode.

The amplifier uses the output current and the reference current to produce a control signal, this difference is also known as the error. This signal is then compared against a repetitive waveform to produce a width-modulated signal that is used to control the switch (Figure 4.4b). When the output current approximate to the reference current, the error decreases so as the width of the modulated signal.

Figure 4.5 shows the controller model used for the simulation.

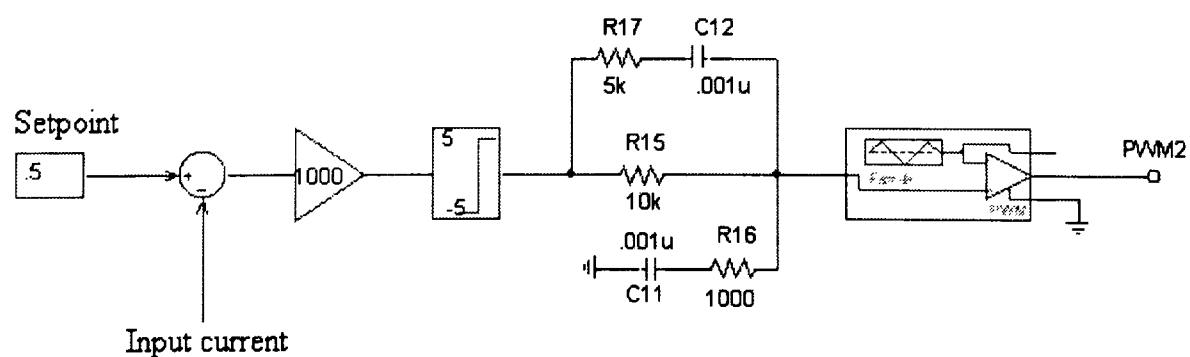


Figure 4.5 Voltage controller model.

The previous controller uses a reference current and the output current. The model normalizes the output difference in order to have a stable PWM waveform. Figure 4.6 shows the complete model for the power converter.

Buck (Step-Down) Converter for Plasma Burner:

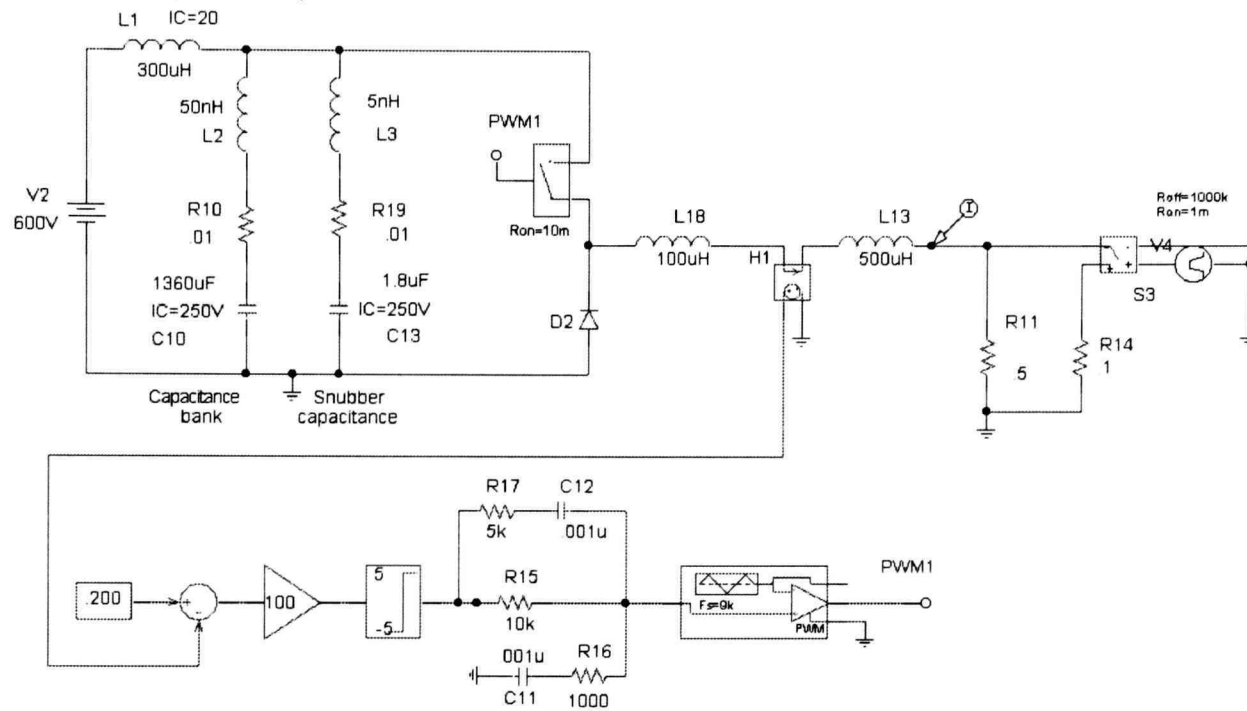


Figure 4.6 Plasma burner.

The load of the model consist of two resistance connected in parallel. One of them is open and it closes at 2ms. This change on the resistance of the load is in order to observe the control for variations on the load. The initial resistance is R11 and then is change into R11//R14. The simulation of the model with a setpoint of 200A is shown on Figure 4.6.

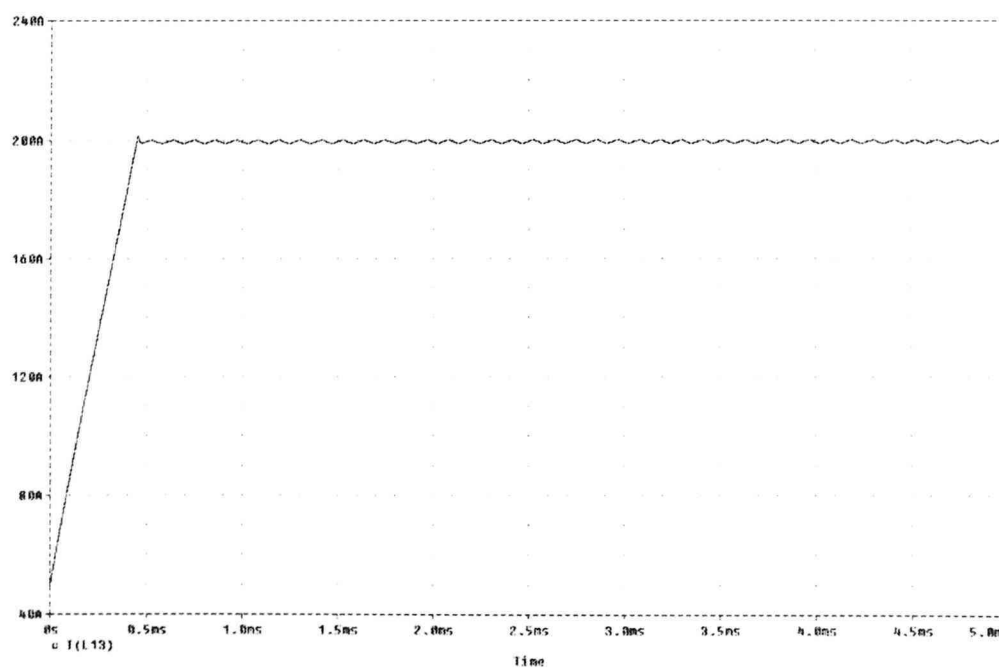


Figure 4.7 Current control with one module.

The error of the system is shown on Figure 4.8, where the error is minimized by the controller.

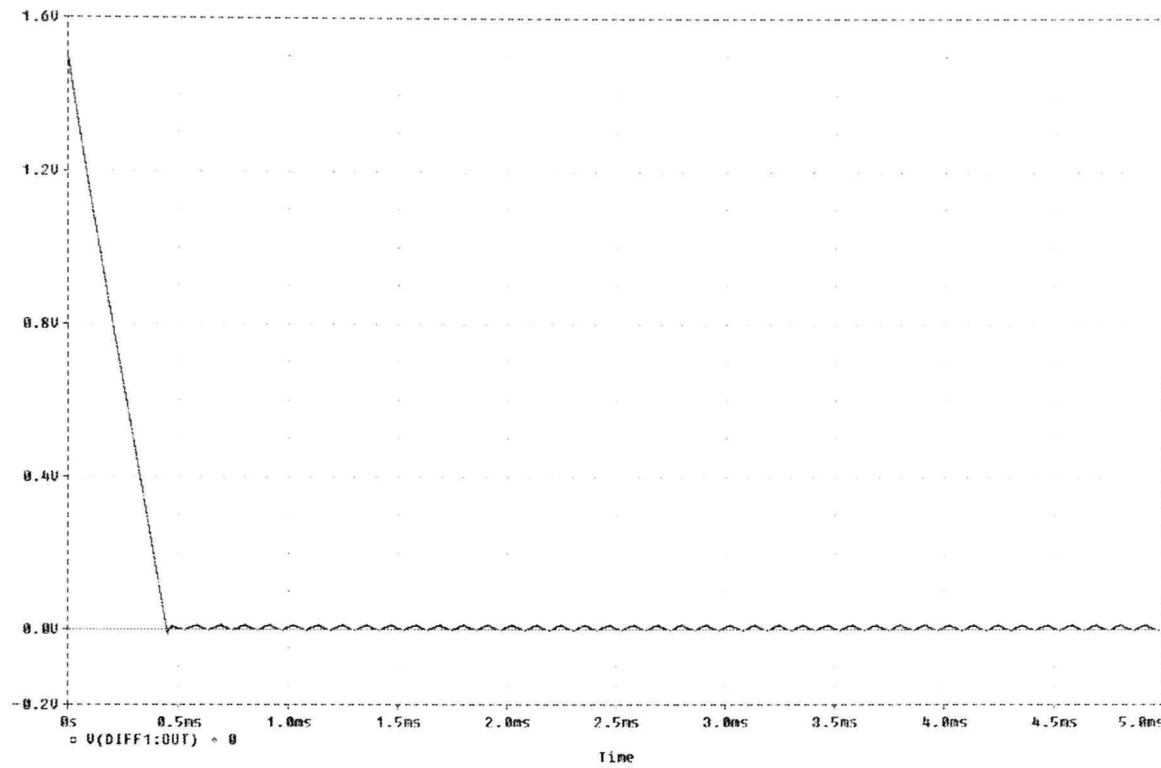


Figure 4.8 Error signal.

It can be observed that the control behavior is very stable for the setpoint current. The next step is to implement more than one controller in order to have more power and to observe the behavior of the load with more than one power converter. Also the purpose of the simulation is to observe the control with more than one setpoint. Another reason to have three independent power converters with three independent setpoints is to protect the modules from each other. In the event that one of the modules fails during the operation, the other two would not have to compensate that loss, therefore suffering a great stress in a short period of time. Figure B.1 in Appendix B shows the converter for three different power converters. It should be noted that in Figure B.1 each model has an inductance at the output and that they are connected to a single one on the load.

Figure 4.8 shows the results of the simulation for equal setpoints and with equal output inductances. It can be observed that the control for this simulation is very stable and that the output current is constant even with the change of the resistance.

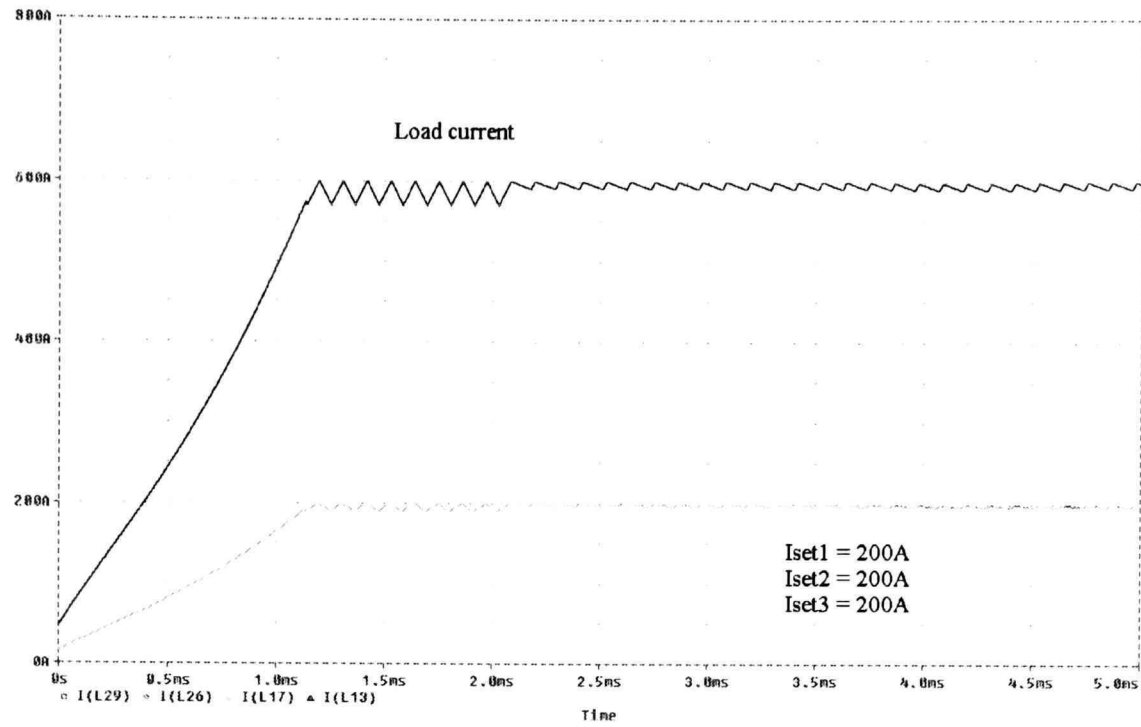


Figure 4.9 Current control with three modules. Iset = 200A

Figure 4.9 shows the simulation for three different setpoints.

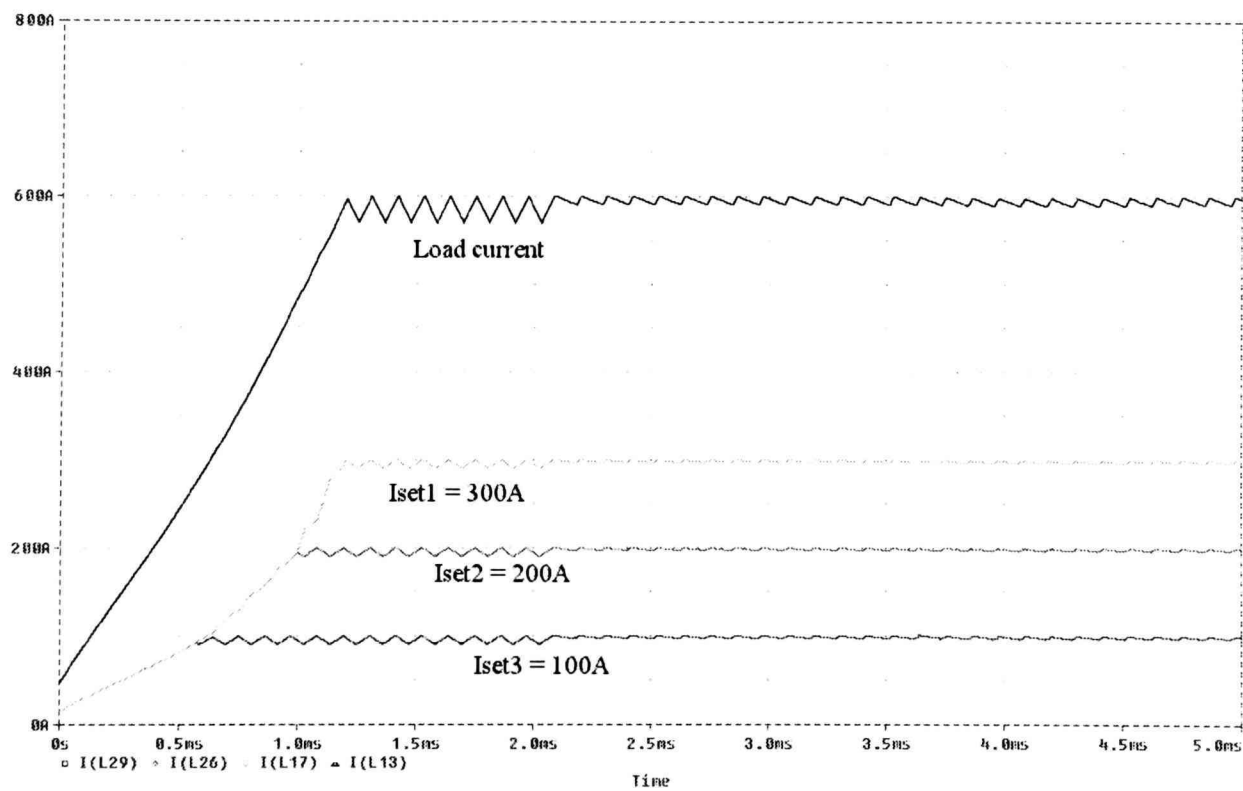


Figure 4.10 Current control with different setpoints.

The result for the previous simulations shows that the controller is stable for different setpoints. This was an important issue, because in the event that one of the

modules could be disconnected abruptly due to a failure, the other modules could still handle the same current without problem.

4.2.2 Current mode

The current control mode differs from the voltage mode on the feedback for the PWM. The UC1823A can be configured for a maximum duty cycle, and in this case was set for a maximum of the 80%. Figure 4.11 shows the model of the second control.

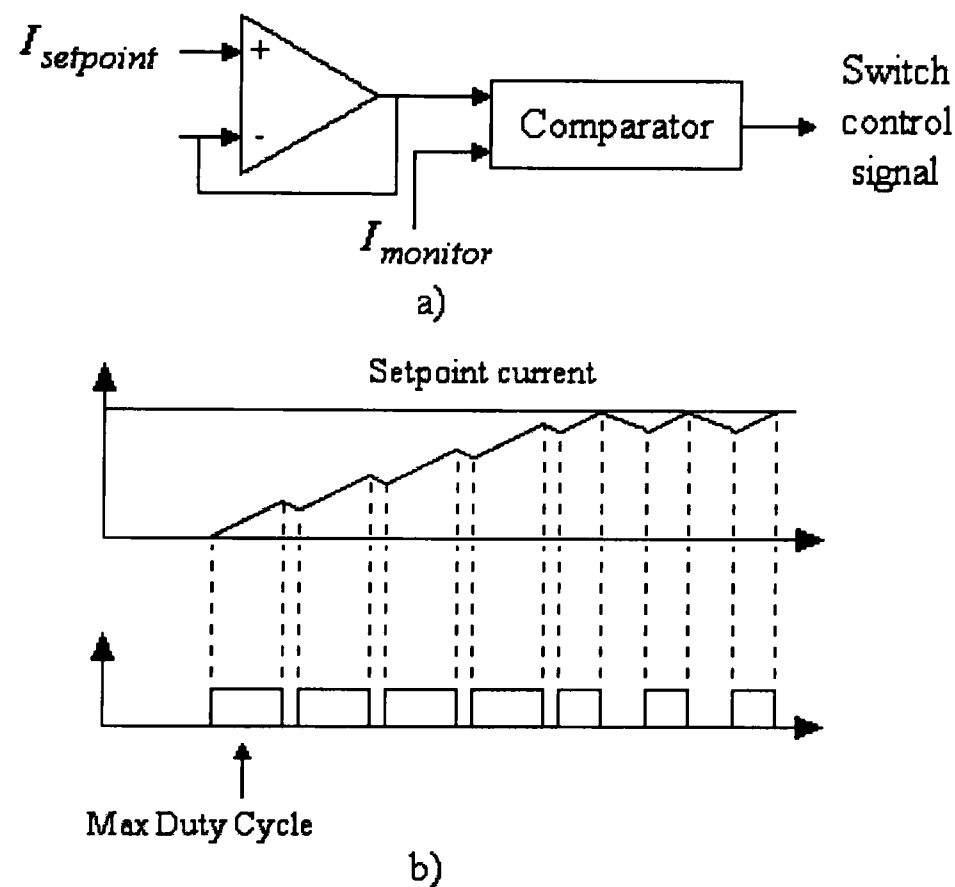


Figure 4.11. Current control: (a) Controller, (b) PWM waveform

The current control for the chip differs from the first on the configuration on some pins. Instead of comparing the difference of the actual current and the reference current, the amplifier is set for unit gain and the actual current is connected to the comparator. Whenever the actual current reaches or 'hit' the reference current, the PWM output turns off and it is turned on until the next cycle. Figure 4.12 shows the controller model implemented on the module using a flip-flop.

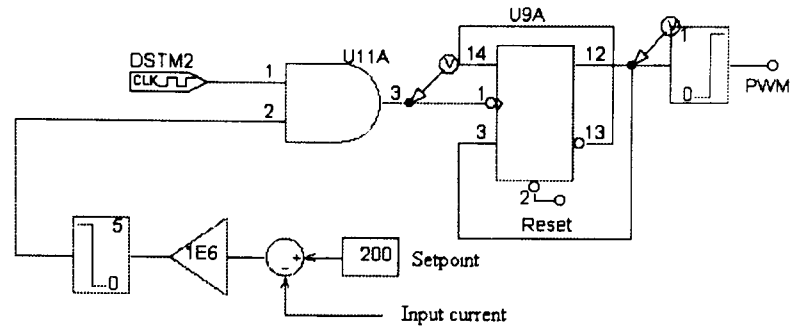


Figure 4.12 Use of a flip-flop as the controller.

The controller for this module was implanted using a flip-flop type J K and a comparator stage as in the first model. The flip-flop is configured to work as a counter that changes the state of the output with each falling edge of the clock transition. In order to resemble the operation of the controller, a maximum duty cycle of the 80% is implemented with the use of the reset of the flip-flop. The clock transitions are the basis of the controller. The clock uses two signals to fire the transitions, a single falling edge at the beginning of the cycle and the error or difference from the actual current and the reference current. Figure 4.13 shows the flip-flop control signals.

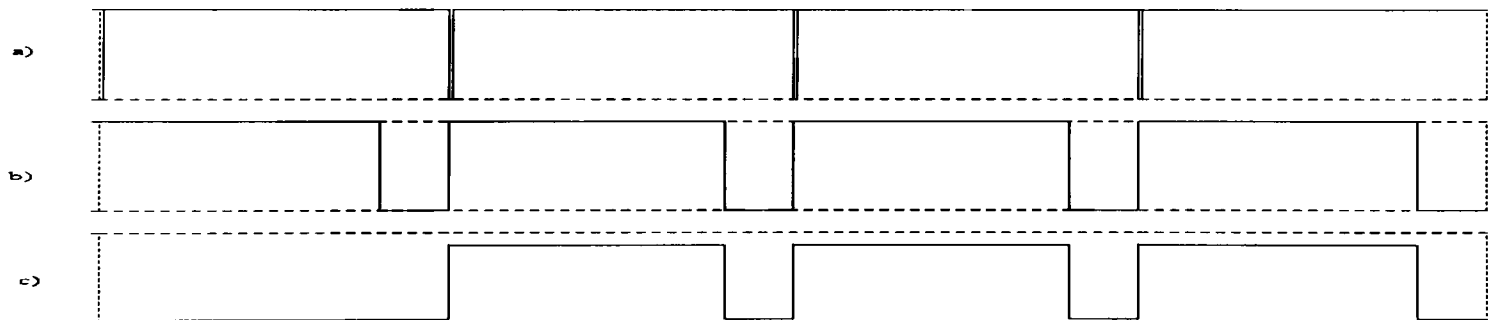


Figure 4.13 PWM control signals, (a) start, (b) reset and (c) clock signal.

In order to turn off the flip-flop whenever the current hits the setpoint is accomplished by an analog to digital interface that consists of the difference between the current and the setpoint normalized to clock signals from 0 to 5 volts. Figure 4.14 shows the model used for a single power converter, and Figure 4.15 shows the result of simulation for a 200A setpoint.

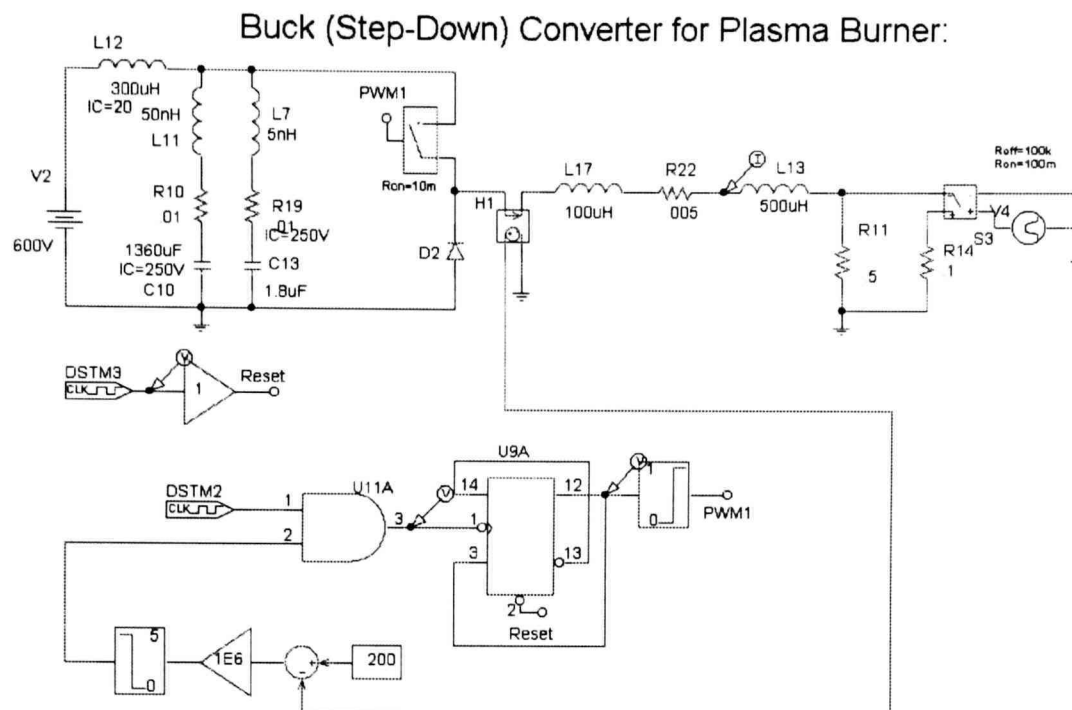
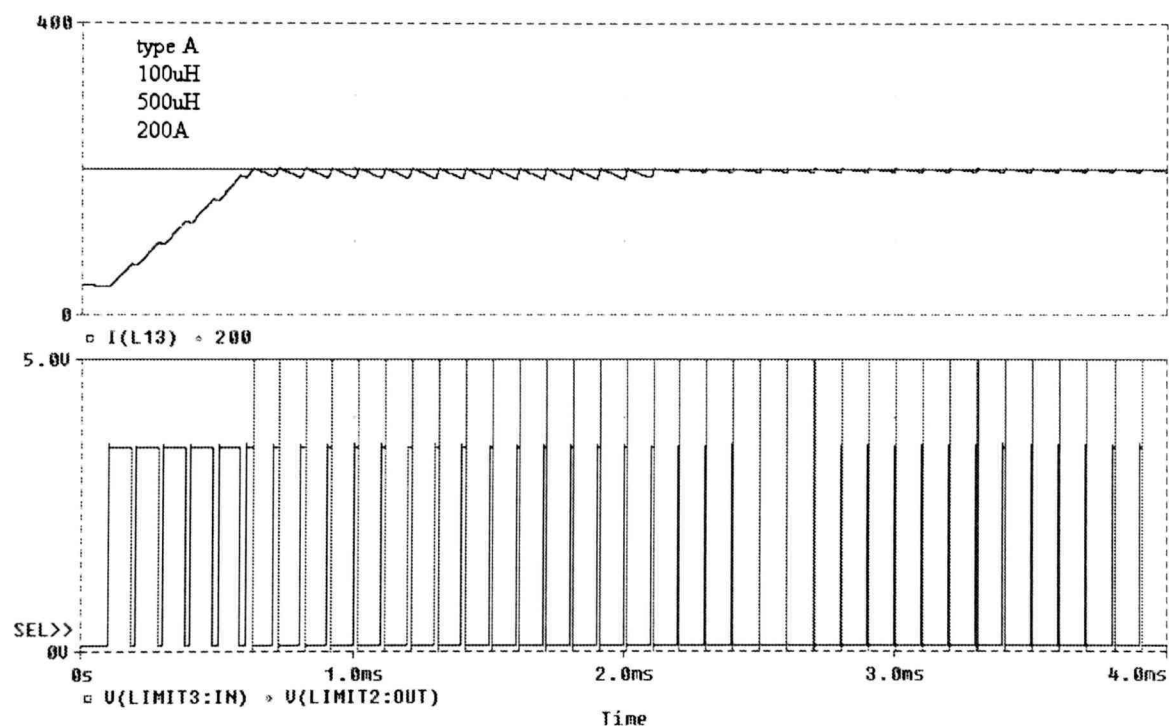


Figure 4.14. Current mode.



We can observe the trigger signal that turns off the controller when the current is the setpoint. This control is also very stable for one module. On the Appendix B is added the controller used for two modules (B.2), where two separate inductances are used to inject the current into a main inductance.

Different cases are simulated, where the values of the inductances and the setpoint varied. The simulations were intended to observe the behavior of the model for

different setpoints and for different values at the inductance. Table 4.1 shows the cases simulated

Table 4.1 Simulation cases.

Figure	L 4	L8	L13	Setpoint1	Setpoint2
4.16	100 μ H	100 μ H	500 μ H	200	200
4.17	100 μ H	100 μ H	500 μ H	100	200
4.18	100 μ H	100 μ H	2 μ H	200	200
4.19	100 μ H	100 μ H	2 μ H	100	200
4.20	100 μ H	110 μ H	500 μ H	200	200
4.21	100 μ H	110 μ H	2 μ H	200	200
4.22	110 μ H	110 μ H	500 μ H	200	200
4.23	110 μ H	110 μ H	2 μ H	200	200

Figure 4.16 presents the case where two modules inject the current into the load. Both modules present the same setpoint and 200A and each module has an output inductance of 100 μ H. The inductance for the load is of 500 μ H. The control behaves pretty stable and the ripple current is very low.

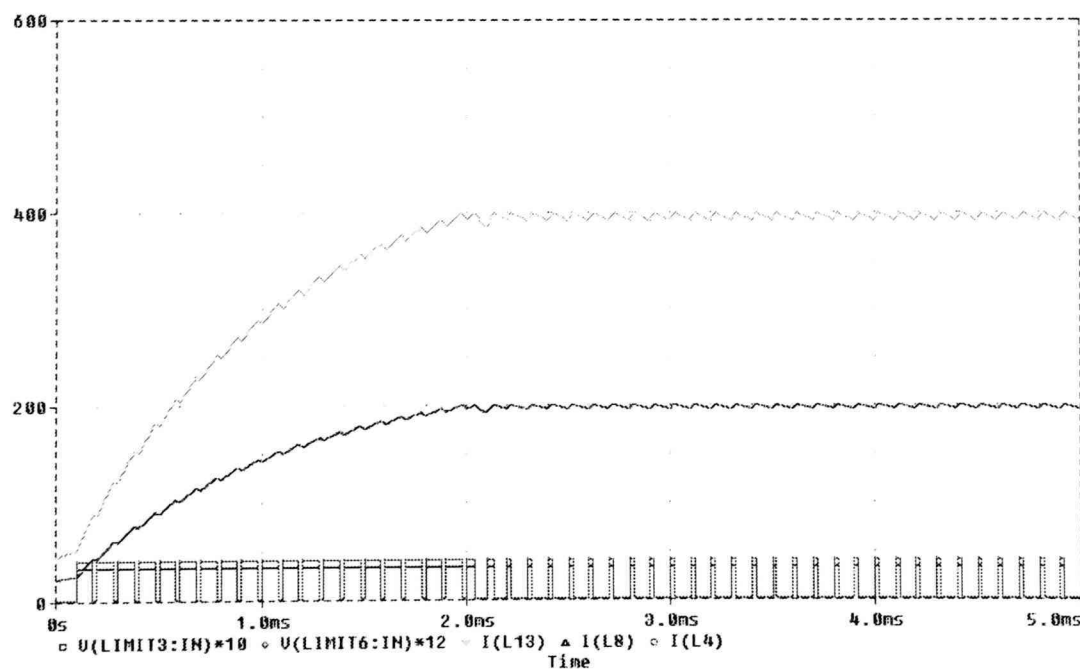


Figure 4.16. Equal setpoints and inductances.

Figure 4.17 presents the case where there is a difference on the setpoints. This case presents an oscillation of the switches due to the difference. Once the module with the lower setpoint reaches the desired current, it turns off, as desired. But when this

happens, the other module presents an increment of current in a short period of time, and the inductor of the module turned off presents a decrement of the same magnitude. This oscillation decreases when the load resistance is reduced. This oscillation is even more notorious by observing the duty cycle of both modules.

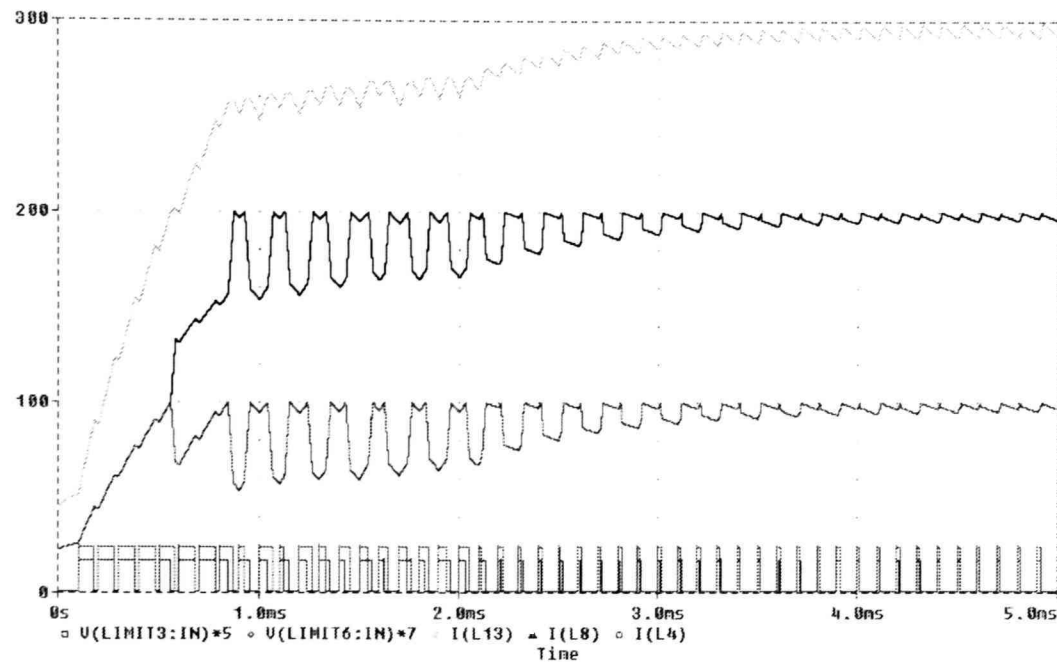


Figure 4.17. Different setpoints, equal inductances

Figure 4.18 present the case where the inductance load is smaller, of $2\mu\text{H}$. This represents a bigger current ripple. The systems is more stable because both modules presents the same setpoint at 200A

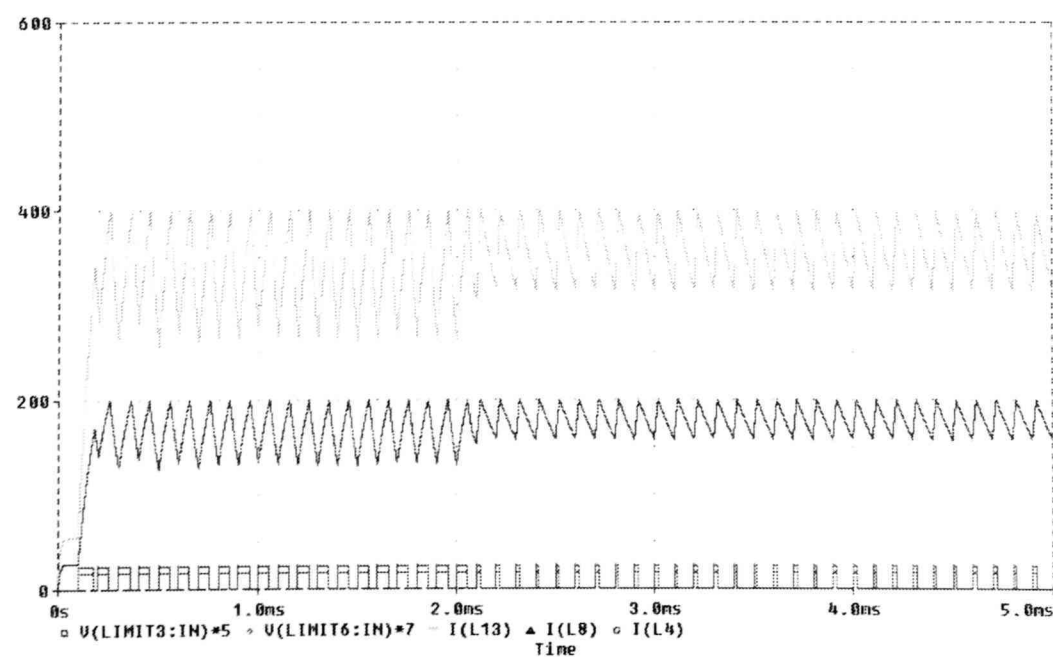


Figure 4.18. Low load inductance

Figure 4.19 present a load inductance of $2\mu\text{H}$ but with different setpoint, at 100A and 200A respectively. The case present the same problems as those results shown previously on Figure 4.15, where the current oscillates due to the difference of setpoints.

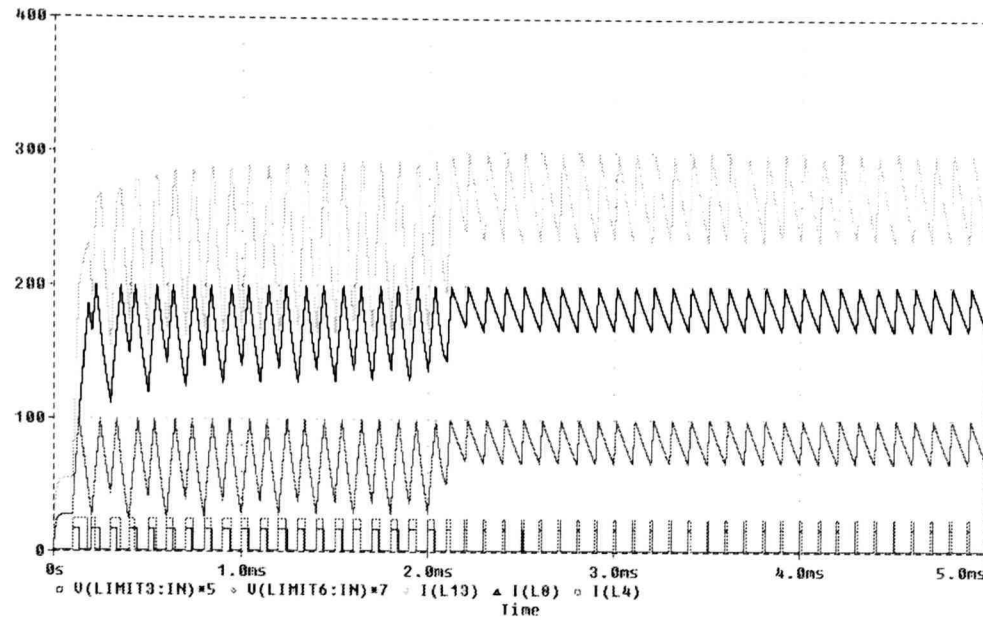


Figure 4.19. Different setpoint, low load inductance.

Figure 4.20 has different values of inductances on the modules. The previous examples have equal inductances at the output of the switch. Now there is a difference of $10\mu\text{H}$ between them. The inductance of the module 1 is $100\mu\text{H}$ and the module 2 is $110\mu\text{H}$. Both models present the same setpoint at 200A. Each module hit the setpoint current at different times, and this produces an oscillation as those previous examples where the setpoint was different. Nonetheless, the modules stabilizes after a while.

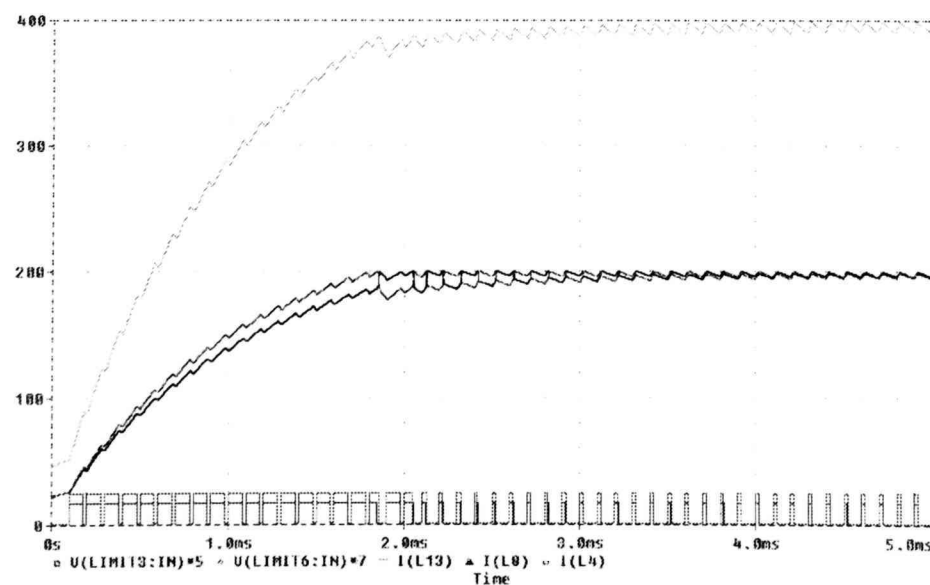


Figure 4.20. Same setpoints, different inductances.

Figure 4.21 has a difference of inductances and also a smaller inductance at the load. The small inductance produce a bigger current ripple which produce oscillations on for the controller.

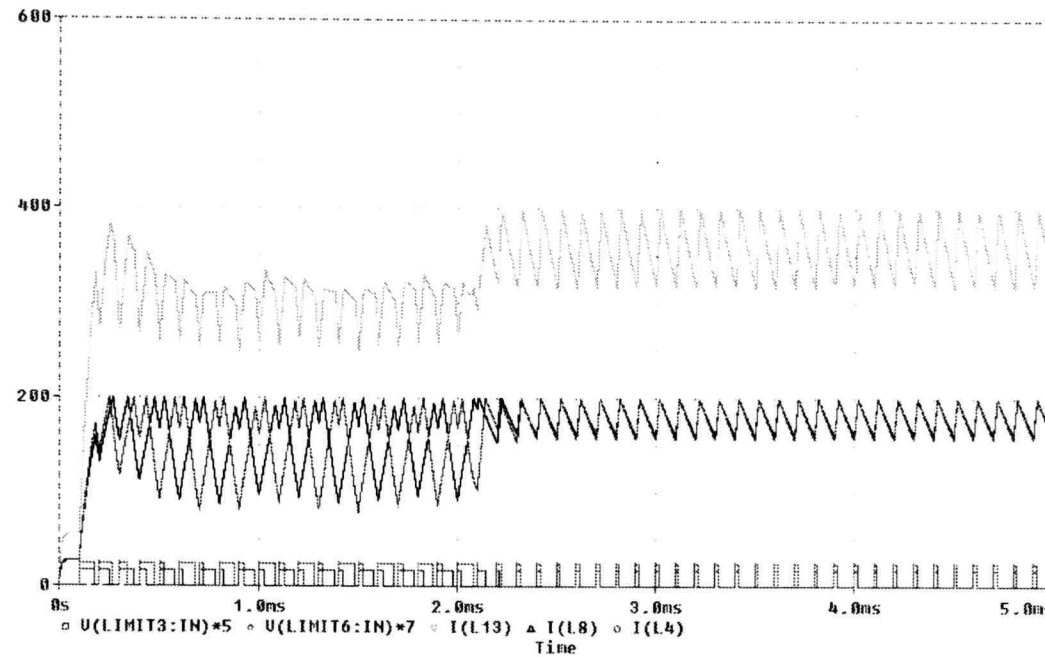


Figure 4.21. Same setpoint, different inductances.

Figure 4.22 shows the simulation for the model with equal inductances of $110\mu\text{H}$ at the output of the switch and equal setpoint of 200A. The result is very similar to the Figure 4.14.

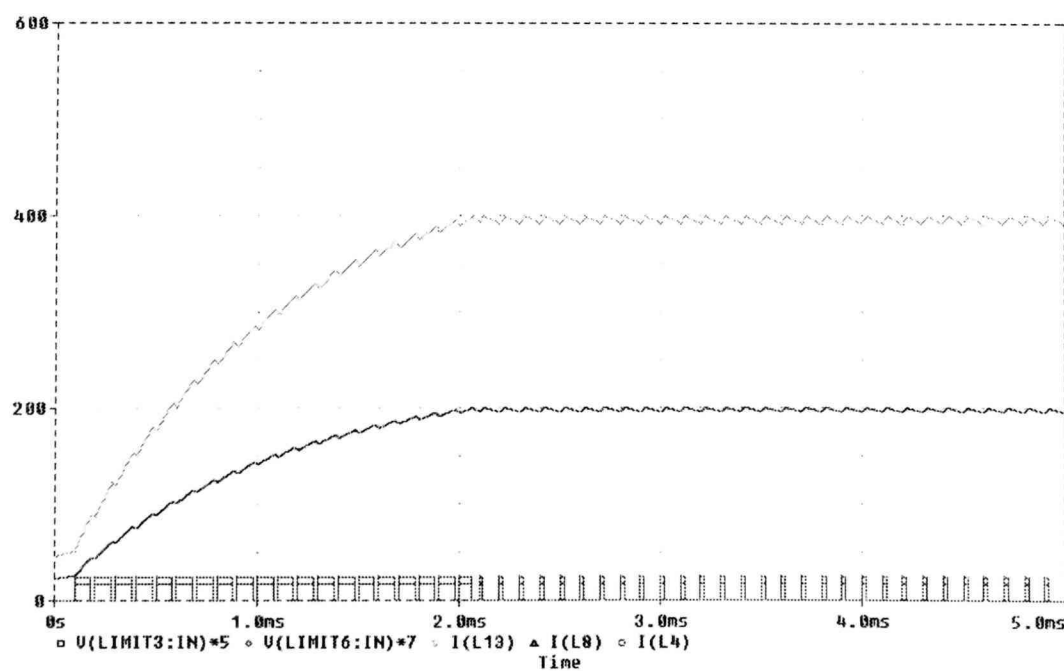


Figure 4.22. Equal inductances, same setpoint.

Figure 4.23 shows the result for the last model, where the input inductances are $110\mu\text{H}$ and the load inductance has a value of $2\mu\text{H}$. The ripple current is high for this case, but the equal setpoint and equal inductances produces a stable control.

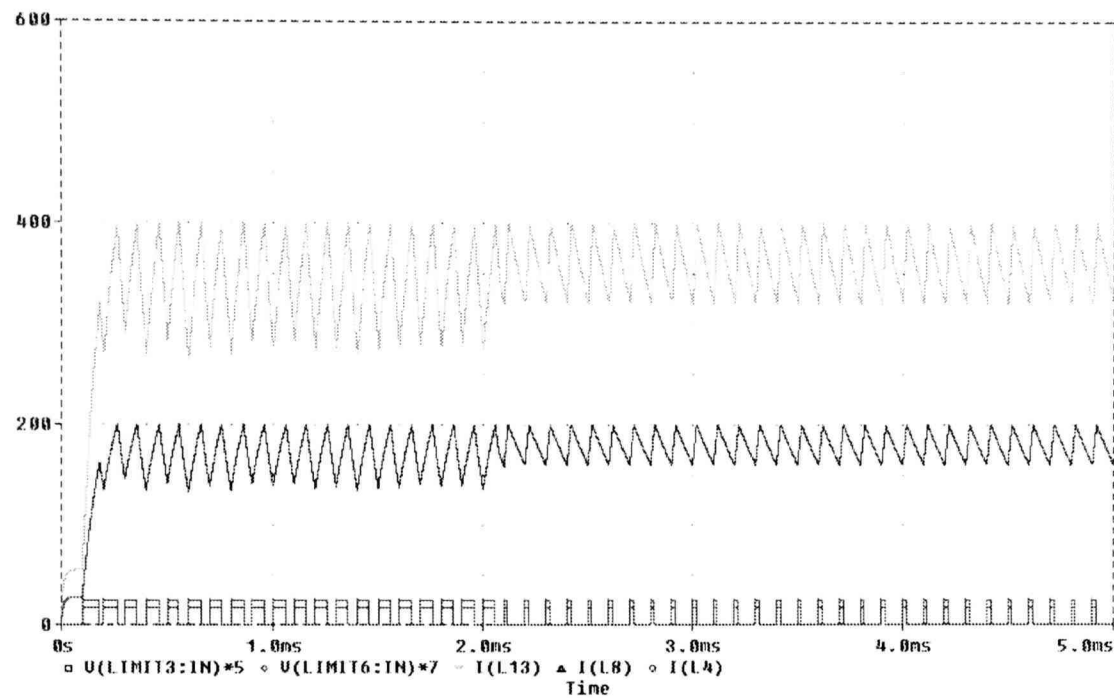


Figure 4.23 Equal inductances, same setpoint, and low load inductance

CHAPTER 5

IMPLEMENTATION

5.1 Electric Diagram

The basic diagram of the power converter used for the torch is presented in Figure 5.1. The power supply is a diesel generator of three-phase voltage with an output of 480 volts. The total power delivered by the generator is 120 kW. The power supply is connected to a transformer with Δ to Y connections in order to have a ground to isolate the equipment of the module. The transformer is connected to the power converter. The power converter has rectifier bridges with capacitors to convert the three-phase ac voltage into a dc voltage. For the controller construction of the power converter, two designs were made. They differ on the control approaches as well as in the hardware used.

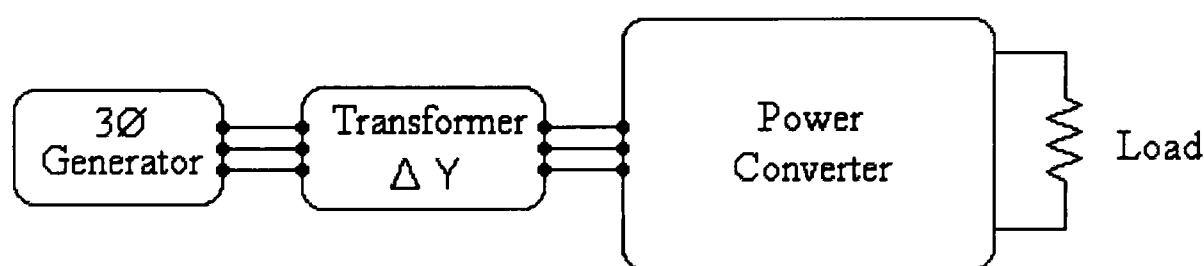


Figure 5.1 Electric diagram.

5.1.1 Components

Components capable of handle high power were needed. Among the various components used for the project are:

- Rectifier bridges,
- IGBT,
- Capacitors and snubber capacitances,
- Inductances,
- Microcontroller,
- Current sensors.

The rectifier bridge used is the model SKD160 from Semikron. This rectifier bridge is capable of handle up to 160 amperes of current. The estimated input current from the power supply was calculated on 450 amps. Three SKD160 units were used in two different configurations. Figure 5.2 shows the rectifier bridge (internal diagram).

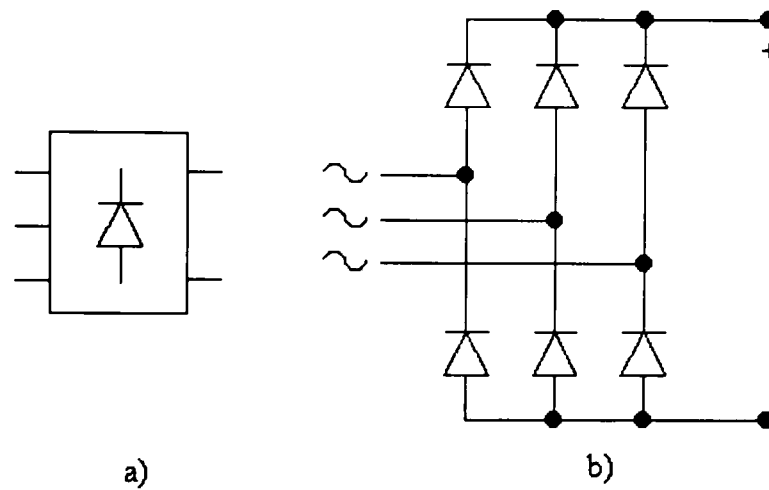


Figure 5.2 Rectifier bridge: (a) symbolic and (b) internal diagram

The switch used on the project was the insulated gate bipolar transistor (IGBT). The IGBT SkiiP 1442 GAR 120 from Semikron was used due to it high capacity to handle current. It IGBT can handle up to 1200 volts and 560 A rms at a maximum switching frequency of 14 kHz. The IGBT is a switch controlled by voltage. The internal block diagram of one IGBT is shown in Figure 5.3. We can observe that this model includes the diode connected to ground, which simplifies the hardware on the buck converter.

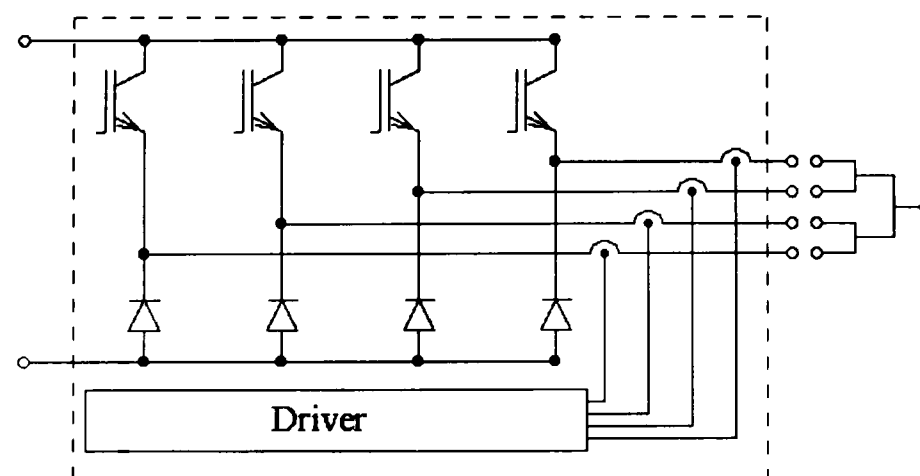


Figure 5.3 IGBT SkiiP 1442. Buck converter

To make the capacitor bank, several models were used, but for the final design the 85uF capacitor from ‘Electronic Concept’ was used. This capacitor can handle up to 1000 volts of dc.

Two different companies fabricated the inductors. Two inductors of 110 μ H were designed by MTE Corporation. Each inductor drives 600 volts of AC or 1000 volts of DC. Another two inductances from Norlake were used. One of them has a value of 100uH and the second one is 500 μ H.

In order to have a ground for the equipment, a transformer made by Mitran in configuration Δ to Y was used. The input voltage is 460 +/- 5% and the output current is 460Y/266 A.

To measure the input and output current, current sensors from F. W. Bell were used, the CLN-1000 and the CLN-300. Each one can measure up to 1000A and 300A, respectively.

5.2 Microcontroller HC12

The MC68HC912B32 (68HC12) microcontroller unit is a 16 bit device composed of standard on-chip peripherals including a 16 bit central process unit (CPU12), a 32-Kbyte EEPROM, a 1-Kbyte RAM, a 768-byte EEPROM, an asynchronous serial communication interface (SCI), a serial peripheral interface (SPI), an 8 channel timer and 16 bit pulse accumulator, an 8 bit analog to digital converter (ADC) a four-channel pulse width modulator (PWM), and a J1850-compatible byte data link communications module (BDLC) [9]. This microcontroller has eight ports from which seven can be configured as input-output.

For this project most of the ports were used as input and output ports, but a basic emphasis is set on to two ports, the PWM module port and the analog to digital converter port. The use of a keyboard and a Liquid Crystal Display were implemented with ports A, B and T.

5.2.1 PWM Module

The 68HC12 microcontroller from Motorola has a PWM module that allows us to drive precise waveforms with hardware instead of software. The pulse width modulator (PWM) subsystem provides four independent 8-bit PWM waveform or two 16 bits PWM waveform or a combination of one 16-bit and two 8-bit PWM waveforms. Each waveform channel has a programmable period and a programmable duty cycle as well as a dedicated counter. A flexible clock select scheme allows four different clock sources to be used with the counter. Each of the modulators can create independent, continuous waveforms with software-selectable duty rates from 0 to 100% percent. The PWM can be programmed as left-aligned outputs or center-aligned outputs [9].

One advantage of the HC12 is that the duty cycle is easily controllable with one register. The generated PWM waveform with the HC12 is as follows. It is a left-aligned signal with a frequency of 10 kHz, a variable duty cycle and has positive polarity. Figure 5.4 shows the resulting waveform.

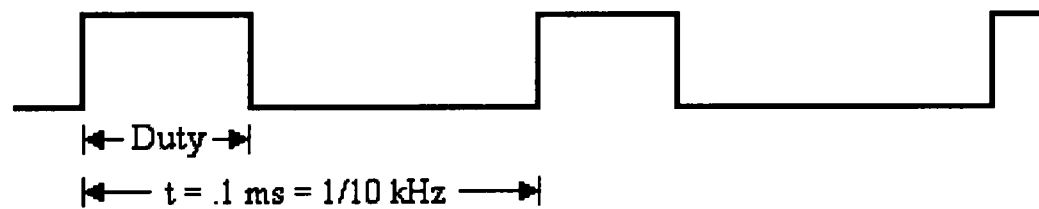


Figure 5.4 Left aligned waveform, positive polarity.

To program the desired frequency, some register need to be declared. As said before, we can have 4 independent channels of 8 bits or two channels of 16 bits. Because the application requires more than 2 outputs, the first option was chosen.

The PWM systems uses the E clock as its primary clock source. From this primary source, it branches into two separate clock chains, clock A and clock B. Clock A drives channels 0 and 1 and clock B drives channels 2 and 3. Both clocks are connected to E through prescaler chains. The job of the prescaler is to divide the E clock. The clock A and B can be connected into a second prescaler, S0 for clock A and S1 for clock B. PWPOL allow us to select the second prescaler. The formula for S0 is as follows:

$$Clock\_S0 = \frac{ClockA}{2 * (PWSCAL0 + 1)}.$$

PSBCK IS BIT 0 OF PWTCL REGISTER.
INTERNAL SIGNAL LIMBDM IS 1 IF THE MCU IS IN BACKGROUND DEBUG MODE.

The diagram illustrates the internal structure of the PWTCL register, which controls four PWM channels (0, 1, 2, and 3). The module is divided into two main sections, each handling two channels.

Channel 0 and 1 Section:

- CLOCK A:** The primary clock for channels 0 and 1.
- Channel 0:** Receives CLOCK A through a MUX to produce PCLK0.
- Channel 1:** Receives CLOCK A through a MUX to produce PCLK1.
- 8-BIT DOWN COUNTER (PWSCNT0):** Receives CLOCK A and outputs a signal to the 8-BIT SCALE REGISTER (PWSCAL0).
- 8-BIT SCALE REGISTER (PWSCAL0):** Receives a value from the 8-BIT DOWN COUNTER and outputs a signal to the MUX for PCLK1.
- CLOCK S0:** A derived clock signal for channel 1, calculated as $CLOCK A / 2$, $CLOCK A / 4$, $CLOCK A / 6$, ..., $CLOCK A / 512$.

Channel 2 and 3 Section:

- CLOCK B:** The primary clock for channels 2 and 3.
- Channel 2:** Receives CLOCK B through a MUX to produce PCLK2.
- Channel 3:** Receives CLOCK B through a MUX to produce PCLK3.
- 8-BIT DOWN COUNTER (PWSCNT1):** Receives CLOCK B and outputs a signal to the 8-BIT SCALE REGISTER (PWSCAL1).
- 8-BIT SCALE REGISTER (PWSCAL1):** Receives a value from the 8-BIT DOWN COUNTER and outputs a signal to the MUX for PCLK3.
- CLOCK S1:** A derived clock signal for channel 3, calculated as $CLOCK B / 2$, $CLOCK B / 4$, $CLOCK B / 6$, ..., $CLOCK B / 512$.

Internal Structure of PWTCL Register:

The PWTCL register is a 16-bit register divided into four 4-bit sections, each controlling a channel. The bits are labeled as follows:

- Bits 0-3 (PCKB2, PCKB1, PCKB0):** Control the clock divider for channel 0.
- Bits 4-7 (PCKA2, PCKA1, PCKA0):** Control the clock divider for channel 1.
- Bits 8-11 (PCKB2, PCKB1, PCKB0):** Control the clock divider for channel 2.
- Bits 12-15 (PCKA2, PCKA1, PCKA0):** Control the clock divider for channel 3.

The register also includes a **REGISTER: PWPRES** field, which is a 4-bit field used to set the prescaler for the PWM module.

***CLOCK S0 = (CLOCK A)/2, (CLOCK A)/4, (CLOCK A)/6,... (CLOCK A)/512**
****CLOCK S1 = (CLOCK B)/2, (CLOCK B)/4, (CLOCK B)/6,... (CLOCK B)/512**

30

The HC12 has the option to select either left-aligned or centered-aligned output. In this project left alignment was selected to simplify the deduction of the formulas to achieve the desired frequency. Figure 5.6 shows the block diagram of the PWM left aligned module.

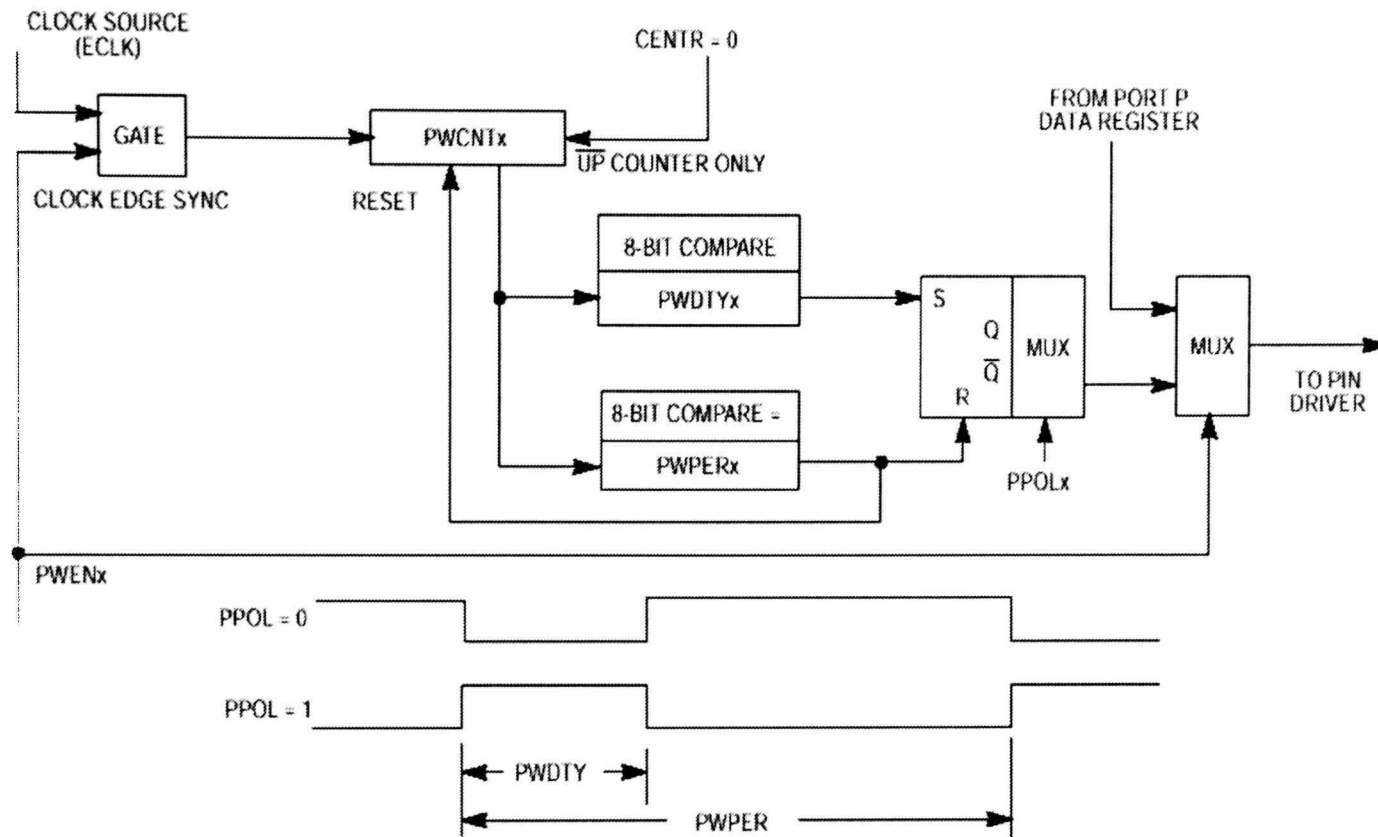


Figure 5.6. PWM left aligned output channel.

The last step it is to set the period of the waveform and the duty cycle. The PWM channel work as follow. The PWCNTx register (counter) will start counting up. When PWCNTx is equal to PWDTYx (duty cycle), the output bit is set accordingly to the polarity, but PWCNTx will continue counting up. When PWCNTx reaches PWPERx (period), PWPERx causes the output bit to be reset, and also causes PWCNTx to be reset to zero, then starts again.

It is important to note that the resolution of the duty cycle depends on the value of PWPERx. The larger the PWPERx value, the more accurate the duty cycle can be adjusted. In order to have this, the period needs to be as large as possible.

For this project, the frequency of the signal was selected to be 10kHz. The clock A and B was set for 2MHz (prescalar of 2 on PWCLK). Clock S0 and S1 were not used. The PWPER set for 200d (out of 255d). The formula is as follows:

$$Clock\_A = \frac{E}{PWCLK} = \frac{8MHz}{4} = 2MHz$$

$$ChannelFreq = \frac{Clock\_A}{PWPER} = \frac{2MHz}{200} = 10kHz.$$

According to the previous formulas, the maximum resolution of the duty cycle is 0.5%. We can consider as a disadvantage the selection of the frequency subjected to the resolution of the period, because we have to sacrifice the period for a precise waveform.

5.2.2 Analog-to-Digital Converter (ATD)

The purpose of the Analog-to-Digital Converter (ATD) is to allow the HC12 to read voltages as 8 bit binary numbers. The HC12 has 8 channels of 8-bit A/D and it is possible to configure them for a 10 bit resolution. Figure 5.7 shows the block diagram for the ATD.

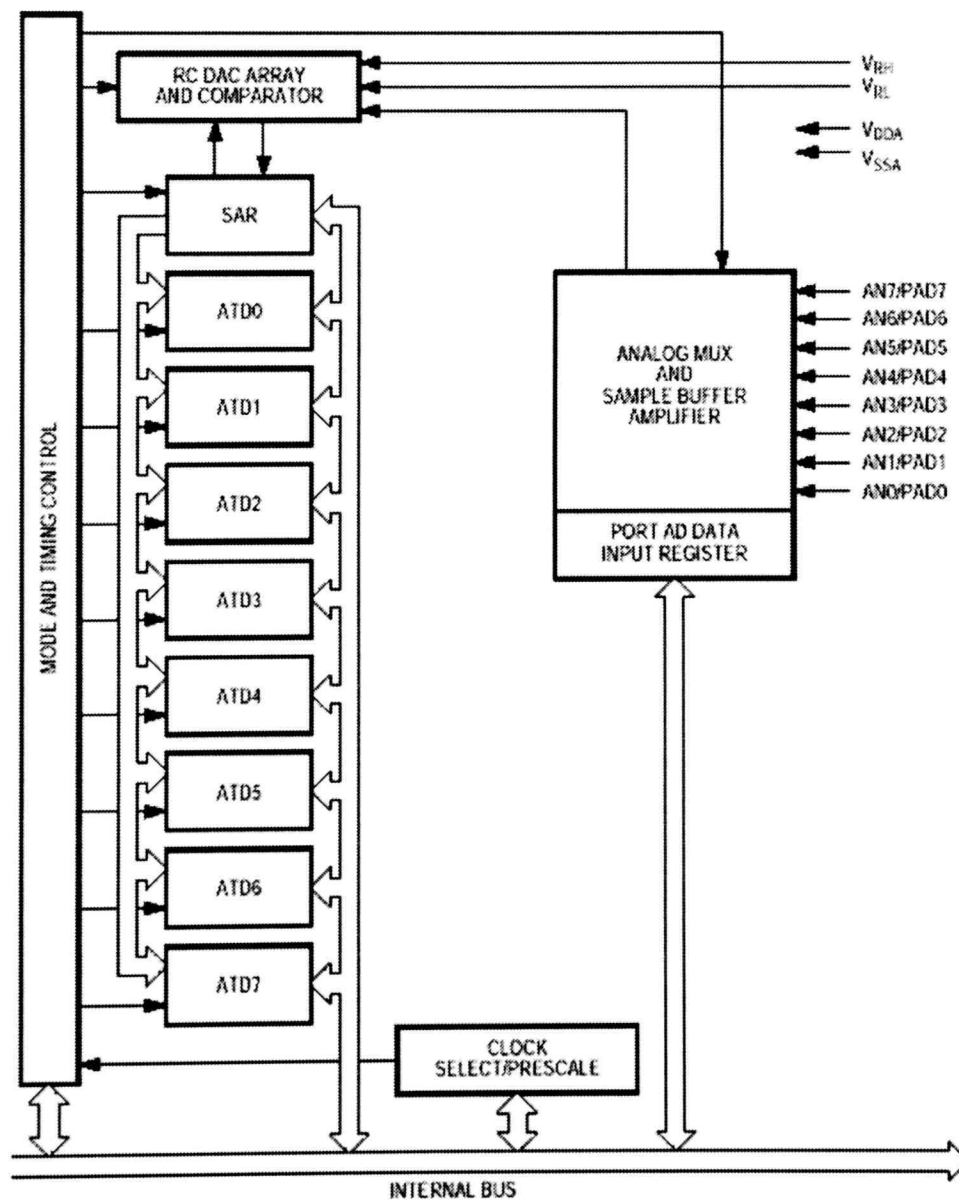


Figure 5.7 Analog-to-Digital Converter.

The module has two modes: single mode and multichannel mode. The single mode takes eight sequential readings from a single channel. The result of the 8 samples is stored from ATD0 to ATD7. In the multichannel mode each input channel is selected and one sample is stored on the result register. The result from channel 0 is sent to ATD0, channel 1 to ATD1 and so on through channel 7. Most of the control bits are in the register ATDCTL5. With the bit SCAN, we can run continuously samples of the channels, each one independent from each other. Bit 4 on ATDCTL is called MULT and set for multichannel mode.

For this project, more than one analog input was estimated, therefore the work focused on the multichannel mode. The bit SCAN was also set to have continuous conversion from the input channels.

5.2.3 Keyboard

Although the keyboard is not a module of the microcontroller, it is explained here because of the connection with the microcontroller. The use of a keyboard is practical as an interface user-system. A common keyboard is the matrix type, which saves the amount of I/O pins used to implement it. Each key has its own combination of row and a column wires. Figure 5.8 presents the internal diagram of the keyboard used on the board.

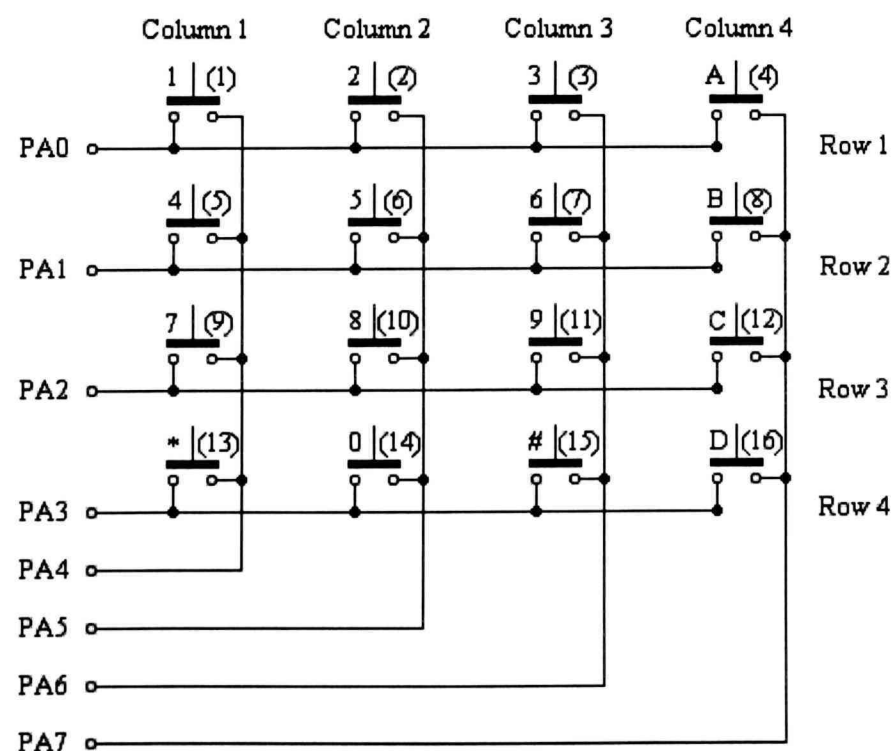


Figure 5. 8. 4 x 4 Matrix Keyboard.

It shows a 16 key matrix keyboard. Each key has a momentary contact switch that is connected to an intersection of a row and a column. When a key is pressed, the switch makes a momentary connection that causes a short circuit between the row and the column. When the key is released, the open circuit exists again between all wires. To determine which key is pressed, the microcontroller must scan the rows and columns to identify the row and column intersection of the short circuit. Each key has an identifying number.

To identify the key code, the microcontroller scans each contact in sequence, and increment a counter each time it scans the next key. When it detects a closed contact, the counter is stopped and it has the corresponding key code. If no closed contact is found, the key code is reset to zero. This process is called *keyboard decoding* [10].

To detect a short circuit, the HC12 drives one of the output lines low and checks a corresponding input line. If it is also low, the key was pressed. If the key is not pressed, the open circuit allows the resistor to pull up the input line into a logic one. For example, to check key code 8, the HC12 drives PA2 low and checks the input at PA5. If PA5 is low, then the key was pressed. To save hardware, the port A has the option to activate a internal pull up resistance, so for the design it was not necessary to set external resistor connected to Vdd. Figure 5.9 shows the diagram used to connect the keyboard to the HC12. It can be noted that also the connection for the reset was set.

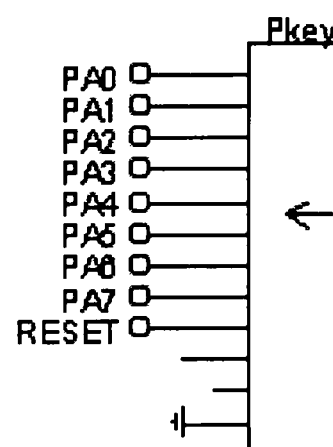


Figure 5.9 Schematic keyboard connections.

The basis of the program for the keyboard was based from [10]. Two main subroutines are developed, getkey and breakkey. Getkey waits for a key to be pressed and

saves the value of the key and breakkey waits for the key to be released. The scan procedure is done using a look-up table. It was necessary to use a debounce routine to avoid accidental bounces from the keypad.

The table 5.1 shows the keys and their code.

Table 5.1. Input keyboard for the matrix board.

Key Code	Row	Column	Port A data (binary)	Port A data (hex)
1	1	1	1110,1110	EE
2	1	2	1101,1110	CE
3	1	3	1011,1110	BE
A	1	4	0111,1110	7E
4	2	1	1110,1101	ED
5	2	2	1101, 1101	CD
6	2	3	1011, 1101	BD
B	2	4	0111, 1101	7D
7	3	1	1110, 1011	EB
8	3	2	1101, 1011	CB
9	3	3	1011, 1011	BB
C	3	4	0111, 1011	7B
*	4	1	1110, 0111	E7
0	4	2	1101, 0111	C7
#	4	3	1011, 0111	B7
D	4	4	0111, 0111	77

Although this is the table used for the key scan, it is important to define the order of the key code for the look up table, because the counter is linked with the symbol of the key pressed. Every time the scan starts, it begins with the counter on zero. If the key pressed is 3, that key will be checked until the counter reaches 3. It s important to observe the order of the values on the look up table. For the keyboard program, there were 4 basic subroutines, keybo, getkey, breakkey and idkey. The listing pseudocode used for the keyboard is show next.

```

; Pseudocode for matrix keyboard driver
keybo:
    call    getkey
    call    breakkey
    return
;-----
getkey
getkey1
    call idkey
        if counter = 16d repeat getkey ;only 16 keys
; The result is saved on ACCA

```

```

        save key
        call idkey
        if actual key = prev key
            then return
            else repeat getkey1
;-----
breakkey
breakkey1
    call idkey
    if counter <> 10, repeat breakkey1
    return
;-----
idkey
idekey1
    keypointer = 0, counter = 0
    portA = *keypointer
    if portA == *keypointer
        return
    increment keypointer & counter
    if counter < maxcounter
        then repeat idekey1
    return
; clear the pointer and the
; counter
; write the key code on portA
; then read portA, if equal, then
; a key was pressed

```

5.2.4 Liquid Crystal Display

Like the keyboard, the liquid crystal display is not a part of the microcontroller. However, it is explained here as part of the modules. For the display of the data acquired on the project it was necessary to use a Liquid Crystal Display (LCD). The model used is the Seiko L4034 4X40 LED Backlight LCD with the HD44780 as the controller chip. The LCD has two controllers since each one control only two rows, and the LCD has four rows. This chip is a special purpose LCD controller; it simplifies the work of the microcontroller. This LCD is a simple but reliable because it requires minimum hardware connections and the rest it is done with software. The communication with the LCD is parallel and requires eight pins to write the commands, two to for the control signal and two more for the clock signal. The LCD requires a control signal to set either characters, the position of the cursor or in general to execute the LCD commands. Figure 5.10 shows the diagram of the connections for the LCD.

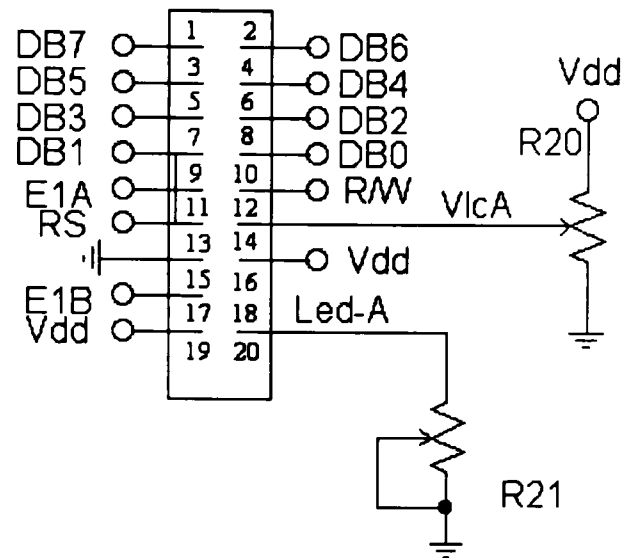


Figure 5.10 LCD Connections

The LCD works through the HD44780 controller, which has two 8 bit registers, an instruction register (IR) and a data register (DR). The IR stores instruction codes such as display clear and cursor shift, address information for display data RAM (DD RAM) and character generator RAM (CG RAM). The DR temporarily stores data to be written into the DD RAM or the CG RAM and data to be read out from the DD RAM or the CG RAM

The DD RAM stores displays data represented in 8-bit character codes. Its capacity is 80x8 bits, or 80 characters for each controller. The DD RAM address is set in the Address Counter (AC) and is expressed in hexadecimal. Figure 5.11 shows the DD RAM address for a 40-character x 4 lines display.

1st controller	1st line	00	01	-----	26	27
	2nd line	40	41	-----	66	67
2nd controller	1st line	00	01	-----	26	27
	2nd line	40	41	-----	66	67

DD RAM address (hex)

Figure 5.11 Address lines on the LCD

The pins DB0 to DB7 write the character or the command to be displayed or executed. E1A and R/W are the control bits for the purpose of the word control. The instruction set for the HD4478 is as show on table 5.2

Table 5.2 Instruction set

Instruction	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Clear display	0	0	0	0	0	0	0	0	0	1
Return home	0	0	0	0	0	0	0	0	1	*
Entry mode	0	0	0	0	0	0	0	1	I/D	S
Display ON-OFF	0	0	0	0	0	0	1	D	C	B
Cursor and display Shift	0	0	0	0	0	1	S/C	R/L	*	*
Function set	0	0	0	0	A	DL	N	F	*	*
Set CG RAM address	0	0	0	1	A	A	A	A	A	A
Set DD RAM address	0	0	1	A	A	A	A	A	A	A
Read busy flag	0	1	BF	A	A	A	A	A	A	A
Write data to CG or DD RAM	1	0	D	D	D	D	D	D	D	D
Read data from SG or DD RAM	1	1	D	D	D	D	D	D	D	D

A best description of each instruction can be found on the technical data of the HD44780. For the execution of the instructions it is necessary to send a pulse of a constant time to the pin E of the controller. The time pulse depends on the instruction. For example, clear display and Return Home require a pulse of 1.6 ms, and the rest only a pulse of 120 μ s. As long as two different LCD were used, and each one having two controllers, four different clock signals were required. To work properly with the LCD, it is necessary to define some bits

- | | | |
|-------------------|---------|--------------------------|
| 1. Display clear | | |
| 2. Function set | DL = 1 | ; 8 bit interface |
| | N = 0 | ; 1 line display |
| | F = 0 | ; 5x7 dot character font |
| 3. Display ON/OFF | D = 0 | ; Display off |
| | C = 0 | ; Cursor off |
| | B = 0 | ; Blink off |
| 4. Entry mode | I/D = 1 | ; +1 (increment mode) |
| | S = 0 | ; No shift |

5.3 First implemented design

The objective of the first design was to control the output power of the power converter. It used IGBTs, capacitors, rectifier bridges and current sensors. Figure 5.12 shows the block diagram for the power converter.

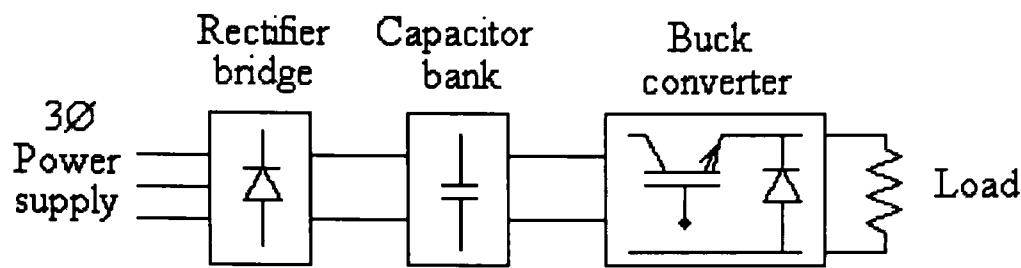


Figure 5.12. Block diagram.

The power converter uses a rectifier bridge and capacitor bank to rectify the 3Ø voltage at the input and IGBTs as the electrical switches for the buck converter. The output voltage is connected to the load. Although the objective of the above configuration was the control of power, at the end it was for the control of current on the load. A more detailed diagram of the above configuration is shown in Figure 5.13. In order to control the buck converter, current sensors and the microcontroller 68HC12 from Motorola are used.

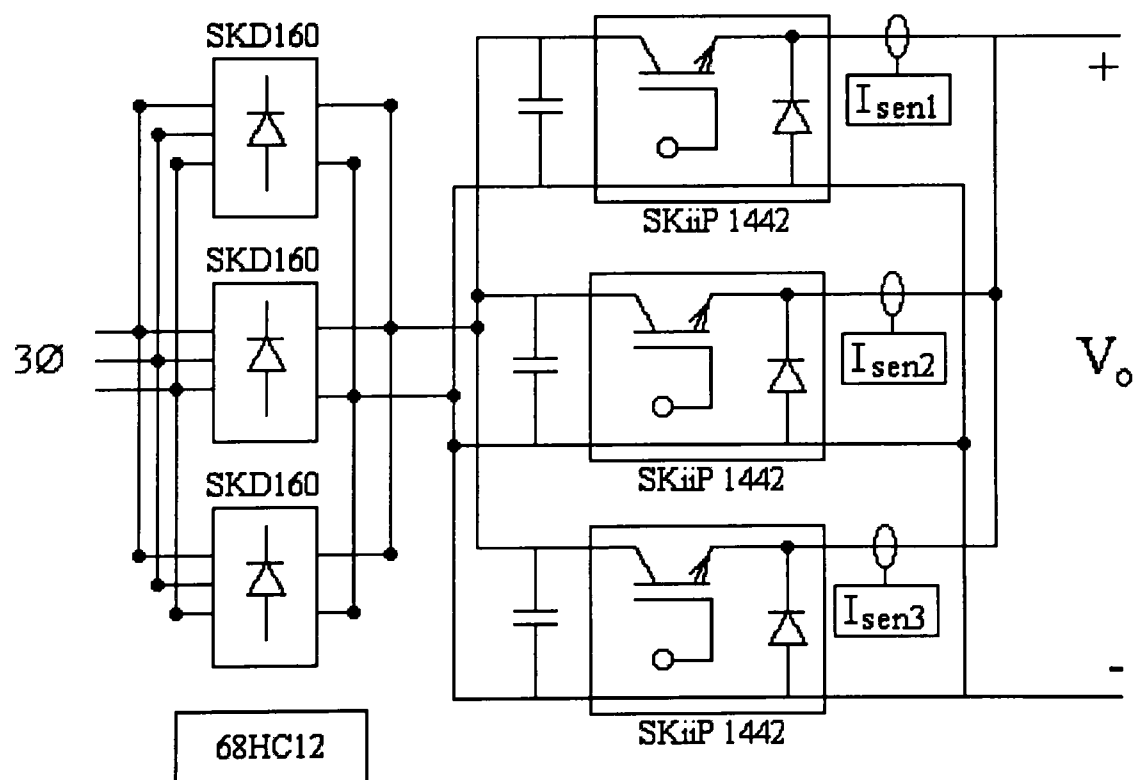


Figure 5.13. First design of electrical diagram

In this configuration, the rectifier bridges and the Buck converters are connected in parallel. The capacitors at the input are used to release stress over the power supply.

The output of each sensor is connected to the 68HC12. For this design, the 68HC12 is the controller of the power converter.

5.3.1 Interface boards

For the control of the power converters with the 68HC12, it was necessary to design and construct two boards to interface the microcontroller with the IGBTs and with the current sensor (Figure 5.14).

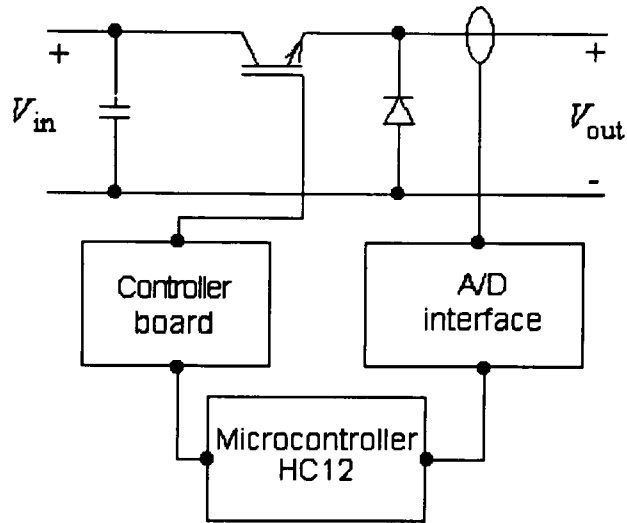


Figure 5.14. Interface board with the HC12

The design of the first board was based on optocouplers. The optocoupler has two purposes: to isolate the power supplies and to adequate the voltage levels from the 68HC12 and the IGBTs. The microcontroller work with voltage levels of zero and five volts and the SKiiP uses voltage levels of zero and fifteen volts. Figure 5.14 shows the configuration to adequate the voltage levels. The optocoupler used are the HCPL2232 from Agilent Technologies.

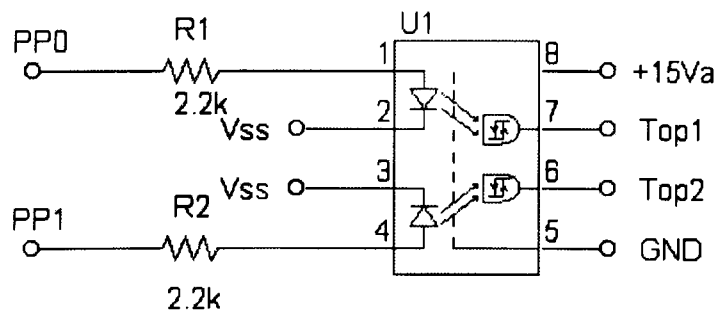


Figure 5.15 Configuration for the optocouplers.

The purpose of this interface was to protect the microcontroller from the IGBT's in case of a failure and to provide an adequate voltage level to fire the IGBTs. The necessary voltage to trigger the IGBT's is 15 volts, and the output voltage from the HC12 is 5 volts. The optocouplers solve the problems, by isolating the voltage sources and also coupling the signals. Figure 5.14 shows the configuration to adequate the voltage levels. The optocoupler used are the HCPL2232 from Agilent Technologies.

The Appendix C shows the complete schematic used for the interface as well as the board for the connections. Figures C.1 and C.4 shows the boards used for the first design.

On the first design, to measure the output current three different current sensors were used. To reject the switching noise the current sensors were filtered with a lowpass filter. The configuration of the filter is presented on Figure 5.15. The filter is a Buttherworth of second order using an operational amplifier TL074A from Texas Instruments. The current sensors were configured for a voltage level between 0 and 5 volts in relation with current levels from 0 to 1000A. For this reason, a unit gain was set for the filters. The cut off frequency for the filter is 100 Hz. This is done to have a signal as smooth as possible.

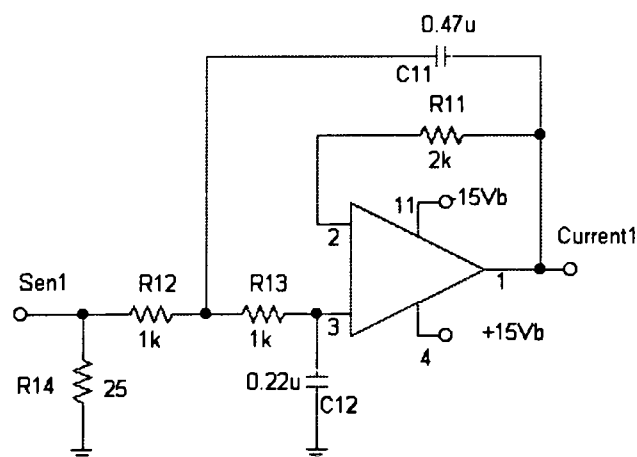


Figure 5.16 Lowpass filter

The Appendix C shows the complete diagram for the analog to digital interface used on the project, it also include the diagrams used for the boards implemented. The design of the filter was simulated on Simplorer, and the response so as the result is shown on Figure 5.16.

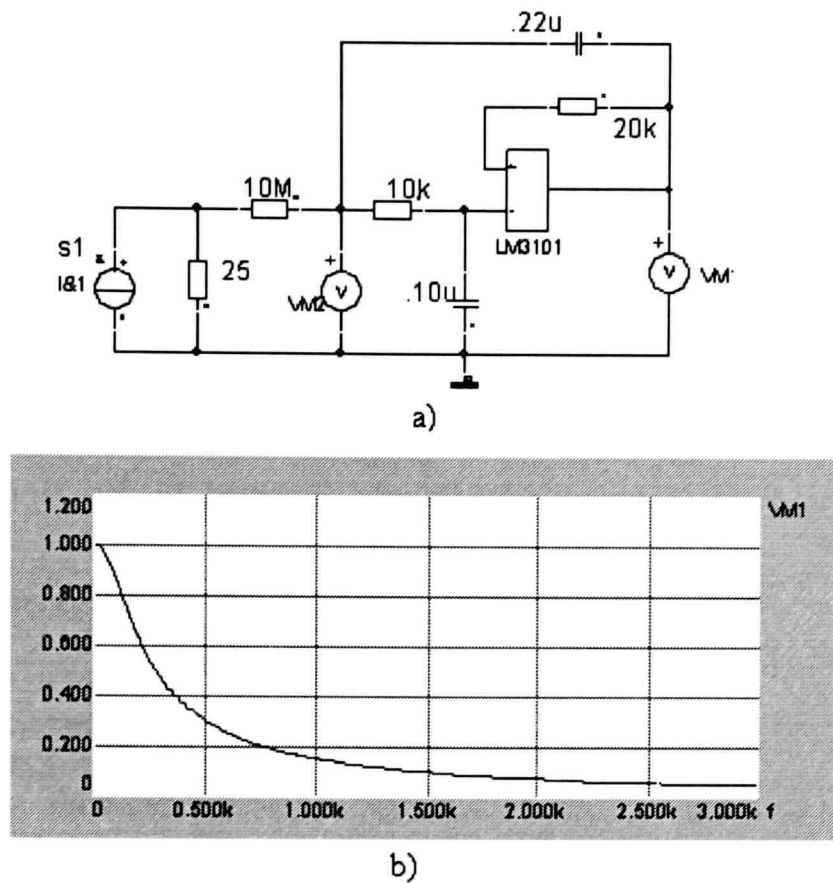


Figure 5.17 Filter design using Simplorer: (a) model; (b) simulation

5.3.2 Control approach.

The design of the controller based on the 68HC12 for the first design involves the use of the PWM module. The techniques described here for the control use a PWM constant signal at 10 kHz where the variable to control is the duty cycle. First the control was intended to be for power. But at the end it focused on the control of the current. A brief description of the controller is presented along with the code and simulations on Simplorer.

5.3.2.1 Increment-decrement control. The first control implemented was an *increment-decrement control*, where the variable to control is the current and the control action is taken on the duty cycle. Figure 5.17 shows the flow chart used for the algorithm. The control takes the value of the current and compares it against the setpoint. Depending on the value, an action is taken, either increment the duty cycle if the current is below the setpoint or decrement if the current is above. This control was used to test the IGBTs in the early stages.

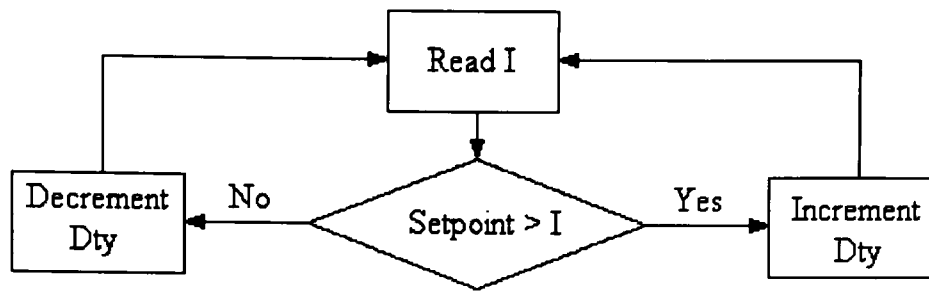


Figure 5.18. Increment-decrement

The pseudocode for the program implemented on the 68HC12 is shown next.

```

; Increment-decrement
control
    define setpoint      :
    read current         :
    if current < setpoint :
        then inc Dty     :
        else dec Dty     :
    repeat control
  
```

For this stage of the project, the 68HC12 test board designed by Dr Giesselman was used. The setpoint was set using the potentiometer of the board.

As previously described, the ATD has a resolution of 8 bits. For this reason the resolution for the current sensor on the HC12 for those voltage provided by the previous configuration is of

$$\frac{1000A}{255b} = 3.90A/bit .$$

As we can see, the resolution is low if we try to keep control over a few amps. At this point, simulations using Simplorer was done for a single branch (Figure 5.18). It can be noticed that there are no capacitors at the output. As long as the load is soil, there is no need to filter the output voltage. The main concern is to control the current on the inductance.

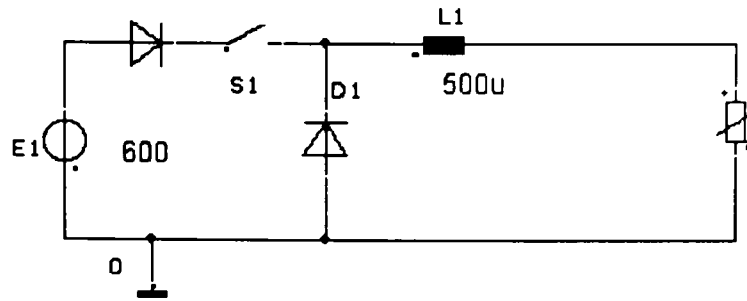


Figure 5.19 Buck converter on Simplorer

The control of the switch is done using a signal called PWM, which is defined as

$$PWM = \tanh(1000 * (Dcon - ST\_rising1)) .$$

PWM generates a width-modulated signal that depends on the value of the controlled duty cycle, *Dcon*. The controlled duty cycle can vary from 0 to 100% on steps of 0.5%, so 200 steps are necessary to reach the 100%. *ST_rising* is a sawtooth generated by a module, and is the signal used to compare the *Dcon*.

The control for the increment decrement was done using a finite state machine and a sampling frequency switch that take one samples each millisecond. The resulting controller is shown on Figure 5.19. The sampling frequency can be varied to change the response of the controller, but in most cases was done with 1kHz. The initial value of *Dcon* can be set from 0 to 100%. There are three states for the control, increment the duty cycle, decrement it, or do nothing. The setpoint of the model, *Iset*, is set by converting the value of the current into a digital value. As long as 255d correspond to a full 1000A on the sensor current, 127d corresponds to 500A, which is the setpoint used on the simulation.

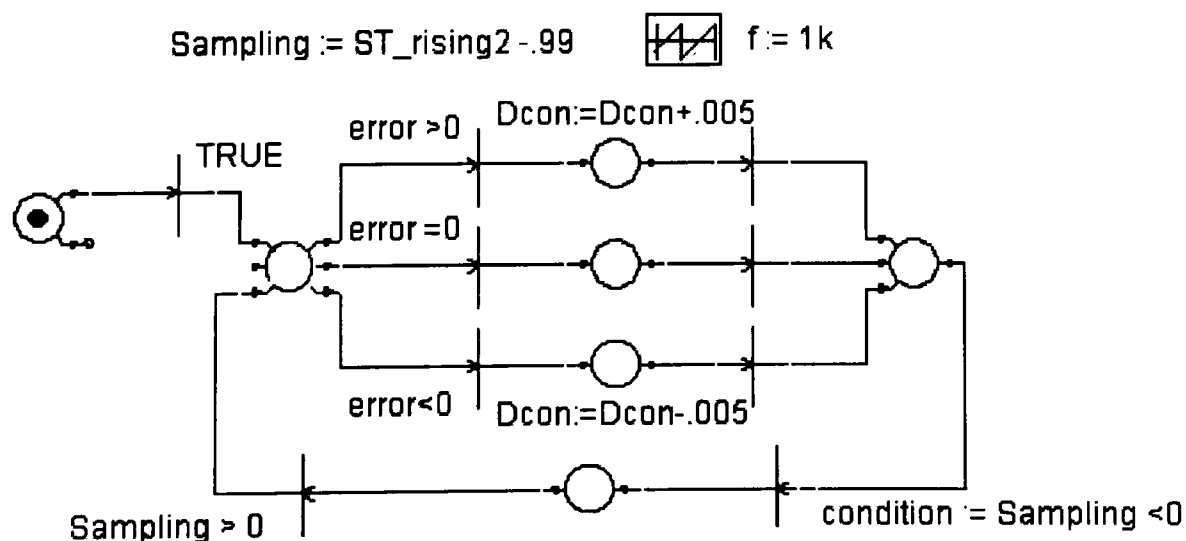


Figure 5.20 Control stages on Simplorer

Two simulations are shown on this section. One of them for a completely discharged inductance and the other with an initial current of 400 A. Figure 5.20a shows the current on the inductor and the duty cycle (multiplied by 1000). Figure 5.20b shows the digital current.

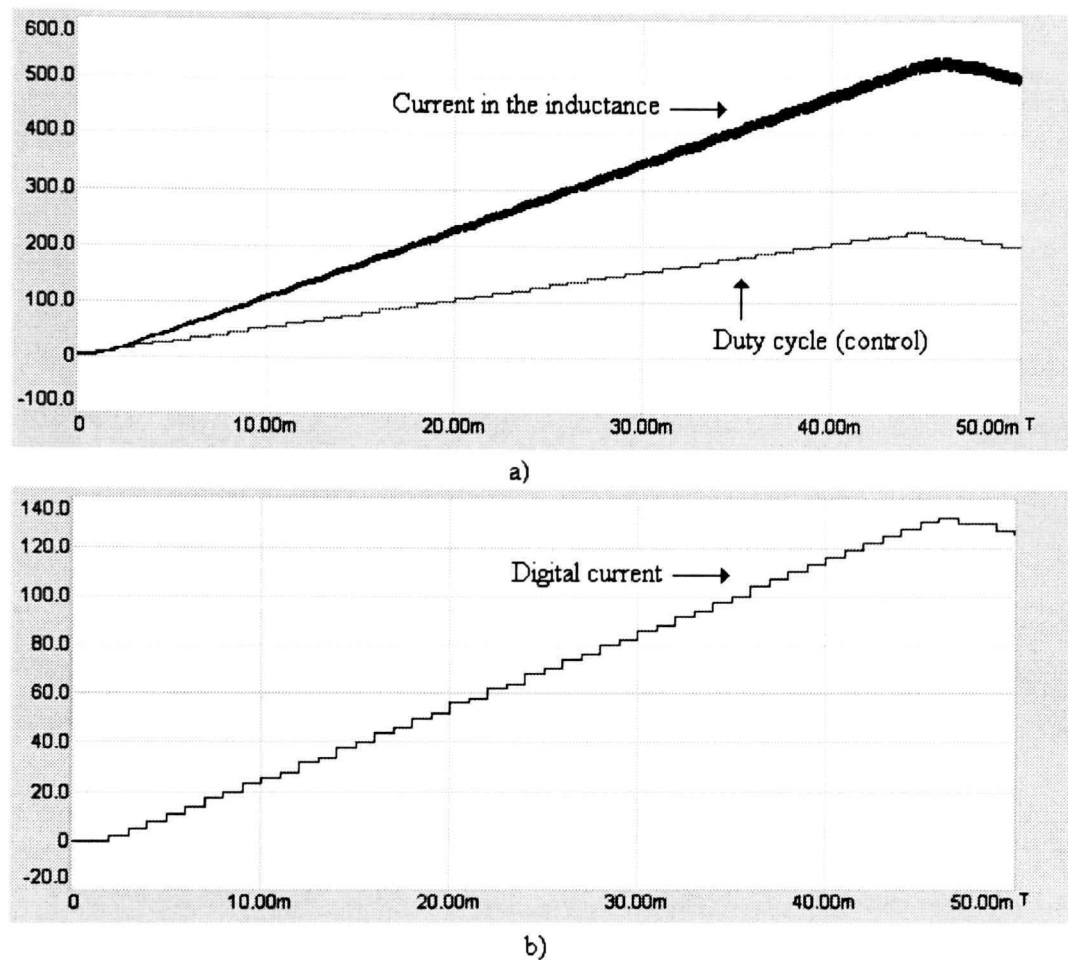


Figure 5.21 Simulation with no initial current on the inductance.

The second simulation is shown next, on Figure 5.21. For this case, an initial current of 400A on the inductance was set, so the control behaves in steady state conditions. For the graphic we can deduct that there is a problem. Due to the inductance, when the duty cycle reaches the desired current, the current keeps flowing, and the duty cycle have to decreased again, but this situation happens all the time so the control oscillates. The practical results show these problems, and moreover, show instability with the use of more than one module.

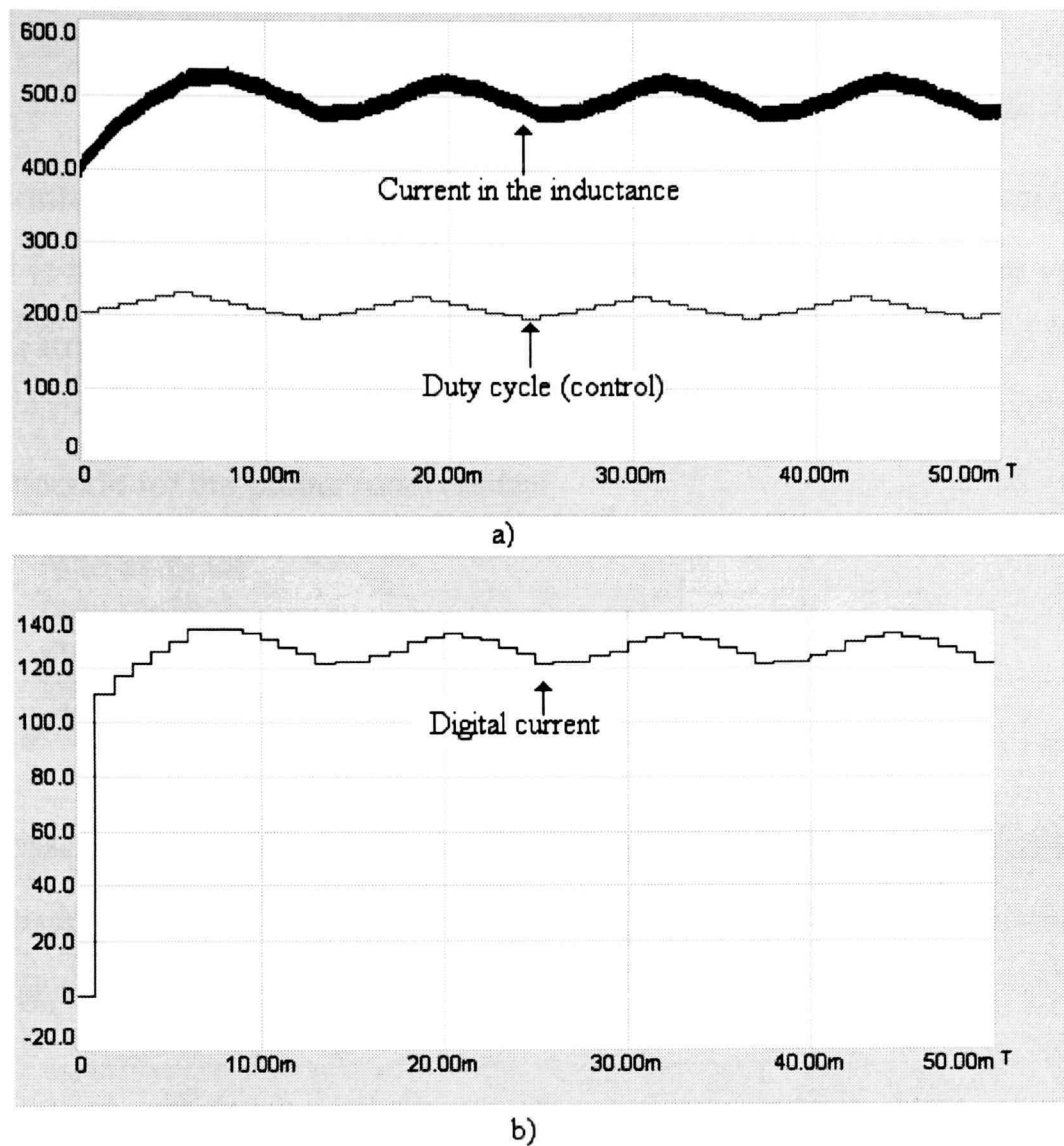


Figure 5.22 Initial condition, $I_L = 400\text{A}$.

5.3.2.2 Proportional control using the 68HC12. The second control approach involves *proportional control*. The control action taken is based on the magnitude of the error or difference between the setpoint and the actual value. Figure 5.22 shows the block diagram used for the control

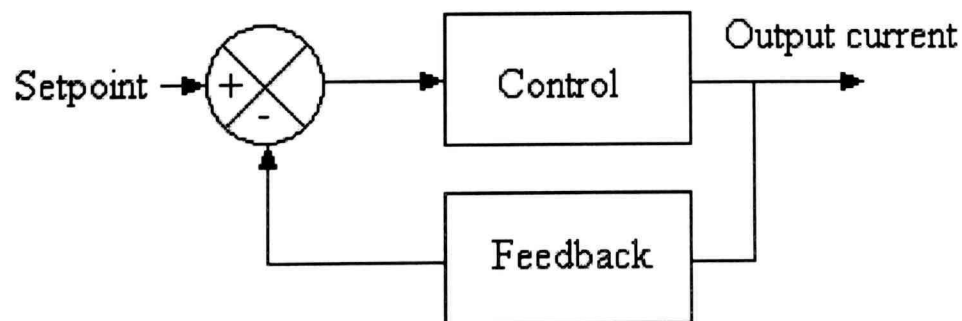


Figure 5.23 Feedback diagram

The formula to control the output current of the module now is as follows:

$$Dty = G * (Setp - L_{current}) .$$

This control involves the use of a gain G , which is the proportional gain. The action control is based on the difference of the actual value and the desire value. The pseudocode to implement this control is as follows.

```
; Pseudocode for the proportional control
proportional
  read setpoint
  read current
  error = setpoint - current
  Dty cycle = gain * error
repeat proportional
```

The approach used in the previous simulation for the sampling frequency was also used in this model, but with higher frequencies. The finite state machine used for this simulation is shown in Figure 5.23.

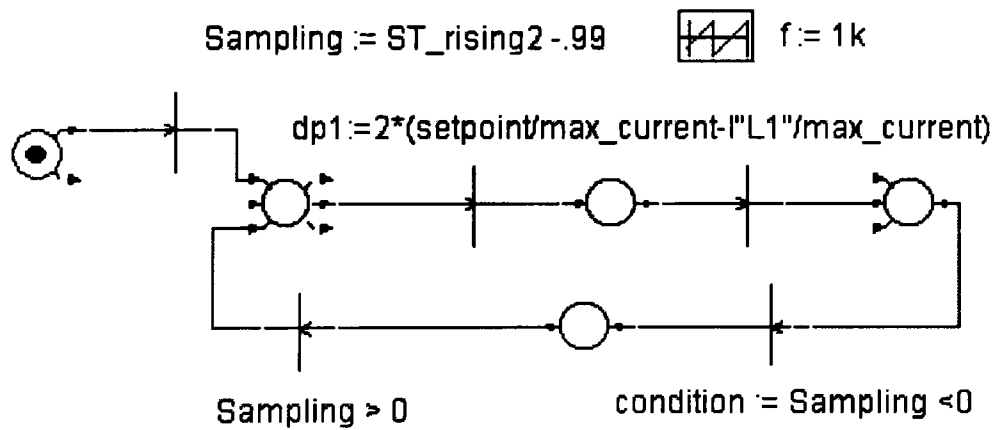


Figure 5.24 Sampling model

In this controller, instead of having three conditions, we have only one that is shown next.

$$dp1 = G * \left(\frac{Setp}{Maxcurr} - \frac{L_{curr}}{Maxcurr} \right)$$

The duty cycle $dp1$ results from the difference between the setpoint divided by the maximum current minus the current on the inductor divided by the maximum current allowed. This difference is multiplied by a gain G , which is on the range from 0 to 5.

It is important to note that due to the limitation on the resolution of the HC12, the values current and setpoint were normalized. The results of the simulation are shown next (Figure 5.24).

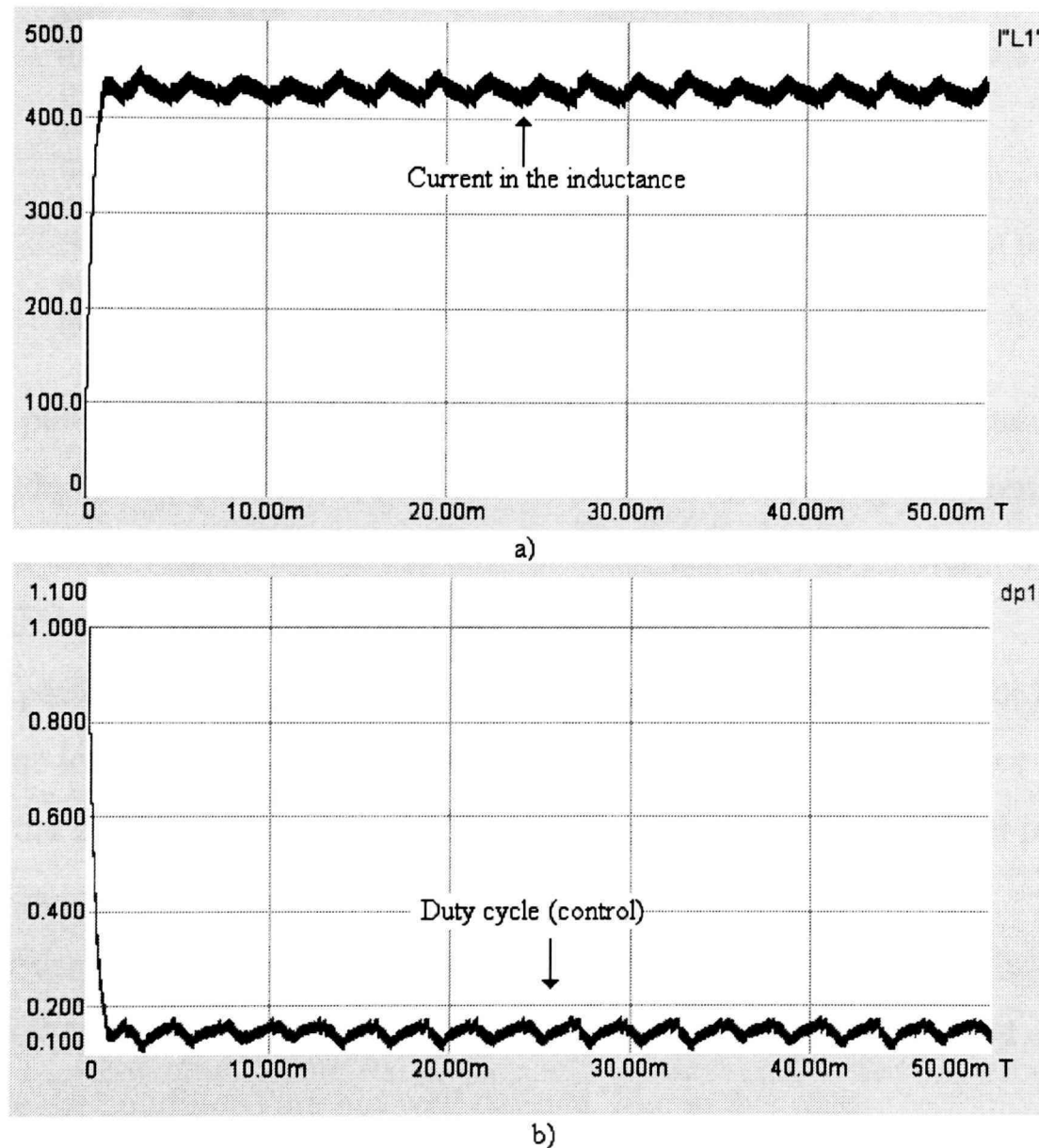


Figure 5.25 Current and Duty cycle.

It has to be noted that for this simulation a higher sampling frequency, of 10kHz, was taken. We can observe the ripple on the current as a product of the sampling frequency of the microcontroller. Nonetheless, the result is better than those observed for the increment decrement control.

On this work, both cases presented noise at the input, even with the analog filters. For that reason, an *averaging filter subroutine* was implemented. The basis of the averaging filter is shown on the next code for the HC12.

```

;   Sensor 1
      clr                               ; 'sumh' contains the total sum
;
;
      ldx    #$093e                     ; defines the first pointer
      ldy    #$093f                     ; second pointer
avg1: movb   1,X-,1,Y-                  ; this is a loop to push the previous values
      movb   1,Y,suml                  ; on the next location
      add    sumh                      ;
      cpx    #$0900                    ;
      bne    avg1                      ; and is repeated 62 times (3Eh)
      movb   ADR4H,1,Y                 ; the actual value is pushed on the top of the stack
      movb   1,Y,suml                  ; and added to the average
      add    sumh                      ;

```

Two pointers push one stack of 32 previous values and then the actual value is saved and added to the previous result. The result is divided by the total number of values added. This process is done like this to have as much accuracy as possible. It is important to note that this process add operation time to the HC12

As a partial conclusion, the HC12 do not represent a good option for the buck converter. The low resolution of the AD module present a poor performance of the system, and the configuration of the IGBTs and the inductance also present problem, that was the reason to take another control approach.

However, it might be possible to implement the control using the fuzzy logic module of the HC12, which has special instructions. This represents a good option for controls were the conditions are not well defined, like in this case.

5.4 Second implemented design

For the second design a new approach was developed. Instead of having the IGBTs and the rectifier bridges in parallel, separate modules containing each one were implemented. Figure 5.25a shows the new configuration for the power converter using three separate modules. Figure 5.25b shows the interior of each module.

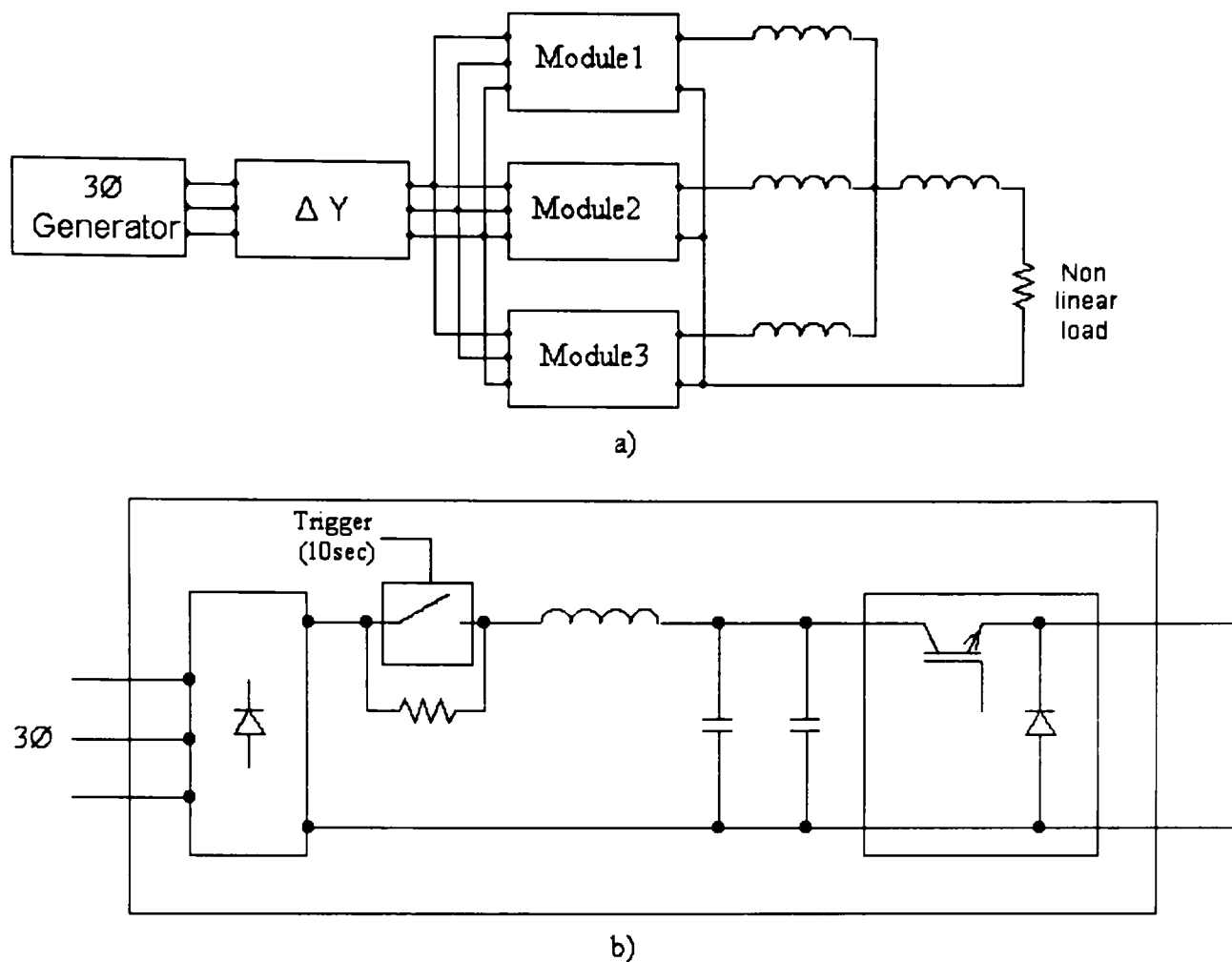


Figure 5.26 Second design: (a) main circuit; (b) interior of the module.

Most of the components from the previous design still are there, but some new were added. The rectifier bridges and the IGBT's are the same. The capacitor bank from the previous design was increased to reduce the spikes current once the switch closed. On a buck converter, the control requires a switch very fast. With such fast open-closed over the IGBT, the power supply can be stressed. To avoid such stress over the power supply, a large capacitor banks is placed at the input of the IGBT, to provide the charge required by the load to avoid the stress the power supply. So for this purpose fast charge-discharge capacitors are required.

The configuration presented on Figure 5.2 had a problem. In order to work, all the IGBT are supposed to work in phase. If one of them applies less current than other, the system gets unbalanced. To avoid this, an extra inductor is placed at the output of the converters modules, and then all of them are connected to a bigger inductance.

The second design also required another approach for the control. This involves the design and construction of a new board. The description of the control will be explained on the section 5.4.2.

5.4.1 Interface board

For the second design, a new board was designed with different purpose. In the previous design, the HC12 was the controller of the IGBTs, and the boards were designed to assist it on that work. Now, a controller board specifically designed for that purpose does the PWM control. Dr. Jim Dickens designed the controller circuit, which also has a multiplexer to monitor parameters from the IGBT modules. The schematic so as the board designed for that purpose is shown on the appendix on Figure C.13.

For this reason, the new functions of the HC12 are to display and to serve as an interface between the user and the controller. The uses for the HC12 are to use a LCD, a keyboard, to monitor analogical inputs from the controller board and to monitor error signals from the IGBTs. The model of the interface between the HC12 and the modules is show on Figure 5.26.

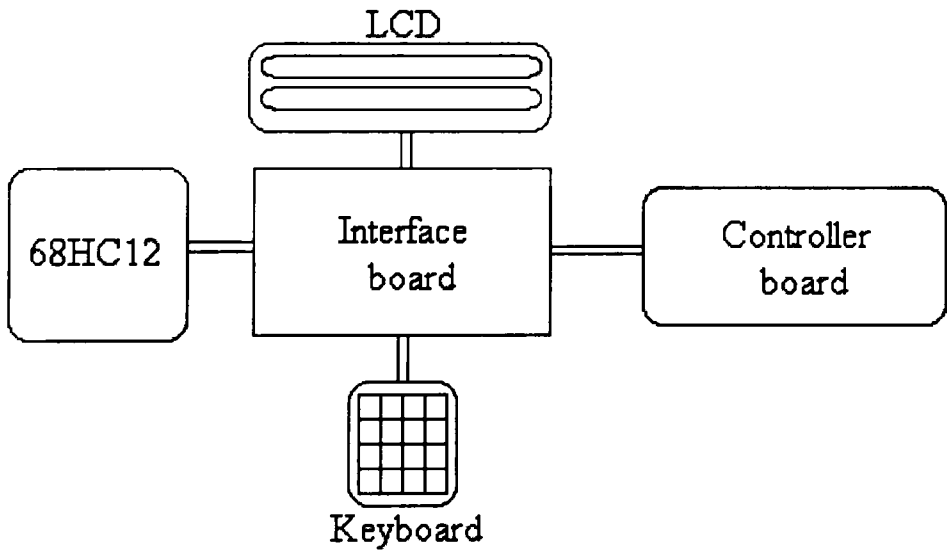


Table 5.27 Second interface

Most of the ports used are for general purposes, even though some of them have an specific purpose, like the ATD or the PWM. The ports used are defined on Table 5.3.

Table 5.3 HC12 ports

HC12 Ports	Function
PortA	Keyboard
PortB	LCD's control signals
PortT	LCD's data words
PortP	PWM output and Analog Input Select
PortAD	Analog inputs
PortS	Overtemperature and Error signals

The keyboard and LCD were connected as explained on section 5.2. The internal diagram for the controller board design by Dr Dickens is shown on Figure 5.27.

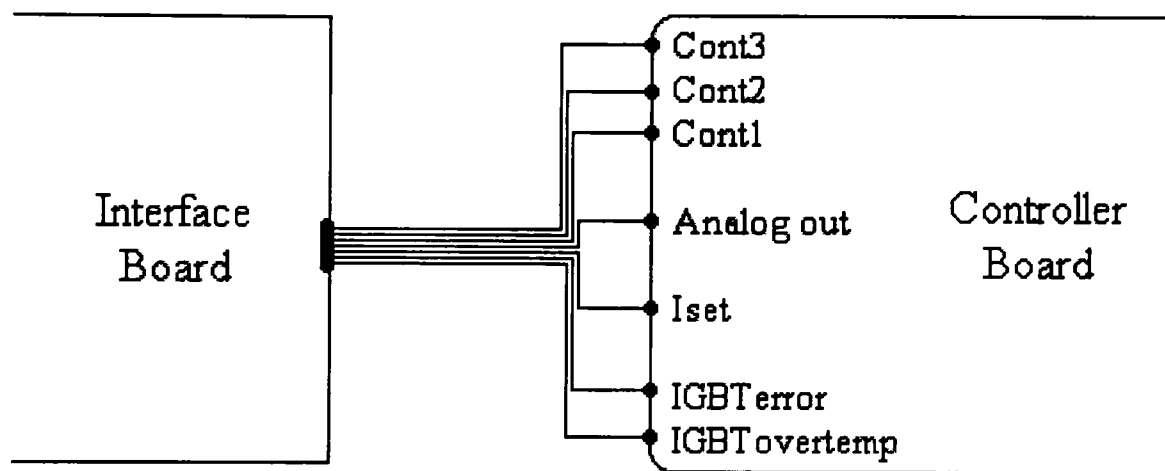


Figure 5.28 Internal diagram of the controller board.

There are seven connections from the controller board to the interface board. The controller board has a multiplexor used to scan the analog inputs from the IGBT's and the current sensors. The multiplexor has 8 analog inputs and therefore use three control signal. The inputs Cont1/2/3 are used for the multiplexor and work with signals of 5 volts. The output of the multiplexor goes to Analog Out, and the voltage goes from 0 to 15 volts. Iset is an input that goes from the HC12 to the controller board. This pin is the setpoint of current of the controller. A voltage from .7 to 5 volts defines the setpoint of current. IGBTerror and IGBTovertemp are output error signals of 15 volts from the IGBT's.

Due to the difference of voltage on some terminals, it was necessary to define voltage divisors at the output of the controller module. The multiplexor can be controlled

with inputs of 5 volts, so the connection was straight. The IGBT error signals were connected through a voltage divisor to low the voltage from 15 to 5 volts. The analog output was also connected to a divisor since the output voltage of the signals varies from 0 to 15 volts. The divisor factor was of 3. Due to the configuration of the current sensors, this represents a problem. The current sensors were configured for a maximum output current of 200 mA on a resistance of 25 ohms. The divisor at the output reduces this voltage level even more thus reducing the resolution of the sensors. For the setpoint the PWM module from the HC12 was used as a digital to analog converter. For this purpose a low pass filter was connected to the output of the PWM channels. The configuration for the connections with the modules is shown on Figure 5.28.

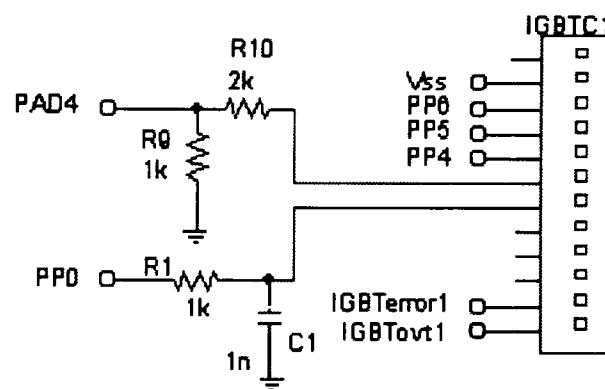


Figure 5.29 Controller module interface.

Because more than one module was used, 4 different input-output connections were implemented on the same board. The design of the board is shown in the appendix Figure E.9.

Based on the last hardware, the new function of the HC12 was to be a setpoint controller. The user have to define the setpoint of each module and type of control, either power or current. There were two basic options for the controller:

- a. Current or power control,
- b. Different setpoint for each module.

Likewise, the HC12 also display information on the LCD. Among the output data are:

- a. Input current,
- b. Output current,

- c. Duty cycle of the PWM controller,
- d. Warning signals of over current and over voltage,
- e. Output power,
- f. Output voltage.

The multiplexer used was the CD4051B from T. I. As explained before, the selection of the channel depend of three bits, cont1, cont2 and cont3. The inputs and the selection word from the multiplexer are listed on the Table 5.4.

Table 5.4 Multiplexer outputs

Name	Device	Voltage range (V)	Cont1/2/3
I_{outmon}	CLN-1000	0-5	000
IGBT I_{mon}	SKiiP 1442, $I_{analogOUT}$	0-10	001
D_{mon}	UC1823A, duty cycle	0-15	010
IGBT temp	SKiiP 1442, $Temp_{analogOUT}$	0-9	011
I_{inmon}	CLN-300	0-5	100

As previously said, the port S is used for the error and over temperature signals from the IGBT's. As long as the new controller chip use a voltage for the current setpoint, the HC12 use the PWM module to generate a dc output voltage using a low pass filter. It is important to note that different setpoints were programmed in order to avoid overcurrent on the modules if one of the IGBT failed. An universal setpoint could lead to malfunctions in the different modules.

5.4.2 Control approach

For the second design the control approach change. Due to the limitation of the HC12 as the controller of the modules, an independent analogical controller was used, the UC1823A from Unitrode Products. Even though two control approaches were simulated, only one configuration for the controller was used. The controller can be configured as voltage control and current control. Even though the configuration on the chip was set as

‘voltage control’, the variable to control was the current. Figure 5.29 shows the control mode for the UC1823A.

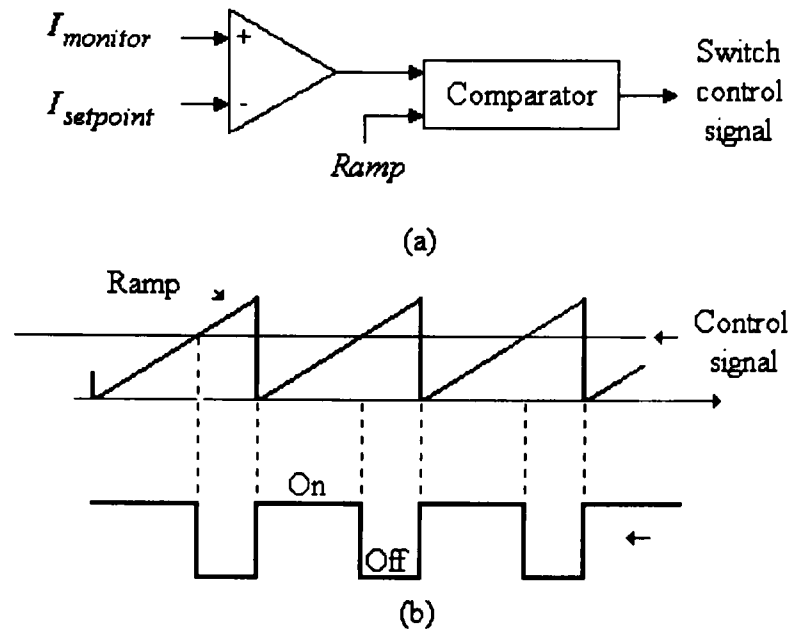


Figure 5.30 New. PWM controller.

The difference between the input current and the setpoint are compared against a repetitive waveform generated by the circuit. The result produces the PWM signal. The simulations for this control were shown on section 4.2.

The control for the modules was done using the UC1823, but the setpoint of the modules was set using the HC12. This can be viewed as a double control, with a regular control using the PWM controller and the HC12 for the setpoint. The block diagram of Figure 5.30 shows the configuration used.

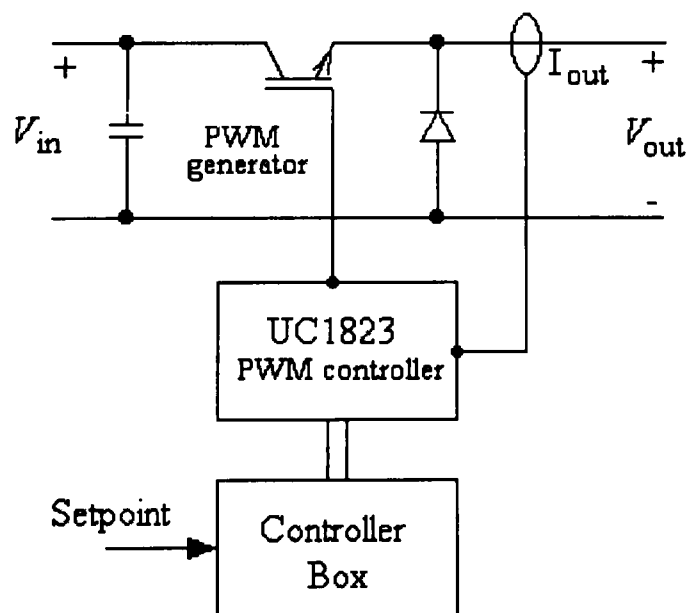


Figure 5.31 Second configuration.

There was two closed loop control on this diagram. The first one was the internal current control from the PWM controller. The second one was the setpoint from the HC12 as long as the user defines a setpoint, and the controller has to match it, the second controller required a lower sampling frequency.

5.4.3 Software

The program used for the control of the LCD and the keyboard so as the controller modules can be divided into main four subroutines, initialization, menu (using the keyboard), display of the data and control loop. A flow diagram is shown on Figure 5.31.

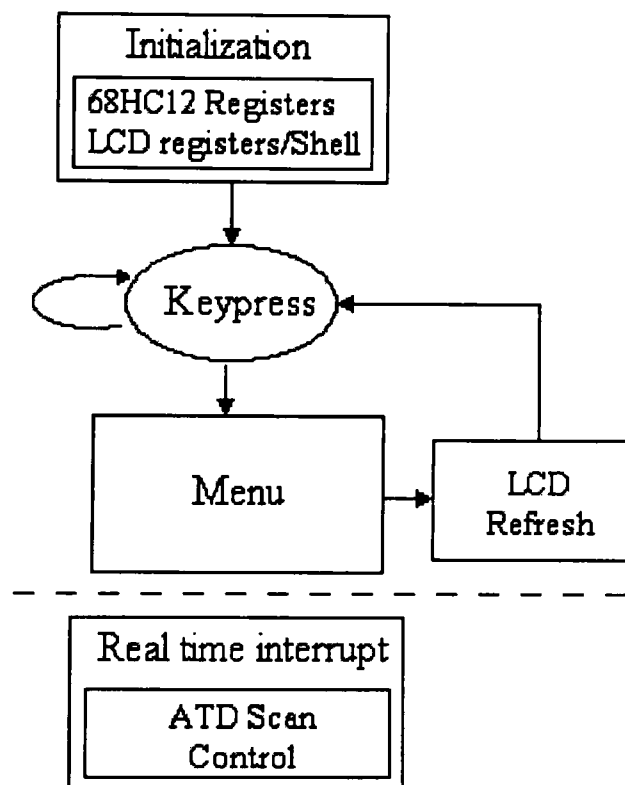


Figure 5.32. Software diagram

The program starts with the initialization of the HC12 registers so as with the LCD registers. At the same time, a 'shell' is written into the LCD. This is the basic core of the LCD's. Then the program continues with the keyboard subroutine. This is a loop where the software waits for a key to be pressed. When a key is pressed, the menu detects the key pressed and executes the command associated to that key. That action may involve a refresh of the data on the LCD. The Real Time Interruption (RTI) is a feature of the HC12. It allows the execution of a subroutine at specific intervals interrupting the

normal code. At the end of the interruption, the program returns to the main routine at the same point it was interrupted. The RTI executes the scan of the ATD module and the control of the modules.

The initialization for the register used to configure the PWM modules, the ATD modules, and the definition of the ports as either input or outputs. The LCD's are also initialized for communication of 8 bits and a 'shell' is also written. Figure 5.32 shows the shell used on the LCD. It is considered a shell because those characters will remain all the time. The values that change will be presented during the operation

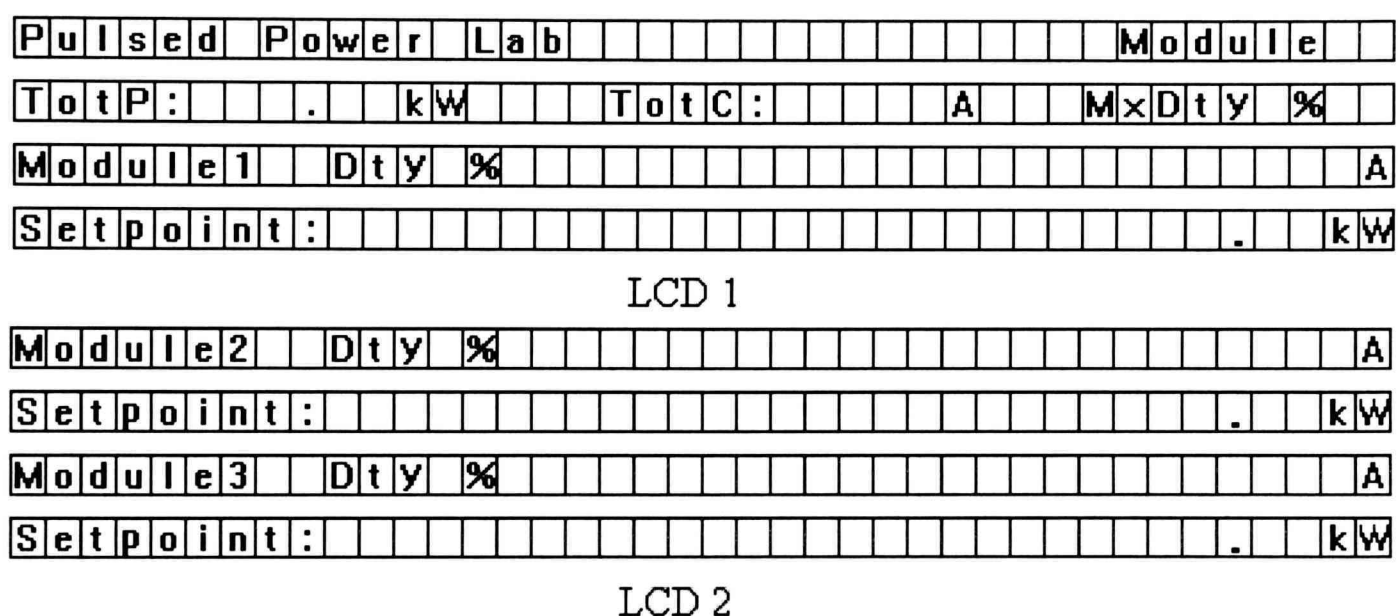


Figure 5.33 Display shell

There are memory registers used as an interface between the keyboard and the modules. If a setpoint is written, then the display has to reflect that change, but also convert the setpoint on the display as a reference value for the control. The diagram on Figure 5.33 shows the relation between the registers. The registers 'dig3', 'dig2' and 'dig3' are used to store temporarily the setpoint of the module in use. When the module change, or the control has to be executed, then the registers are transferred to the setpoint of the module in use, either A, B or C. only a maximum of three numbers can be accepted for 'dig1/2/3'. This is to avoid a shift of a selected setpoint.

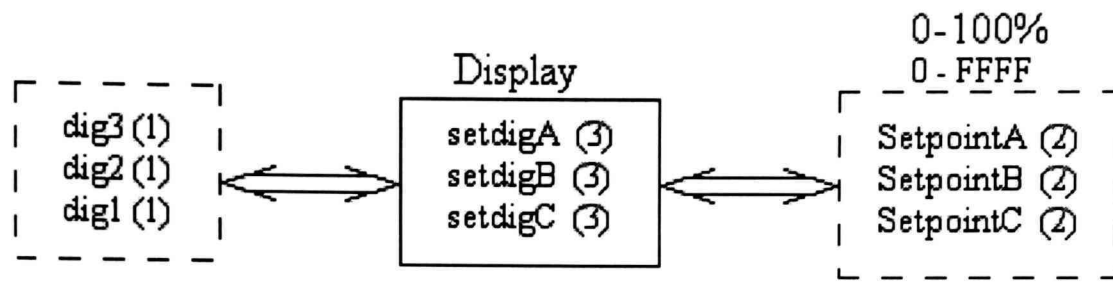


Figure 5.34 Registers relations.

For the menu, the keyboard is used. The keyboard has 16 keys, from which 10 are used for number codes. The character keys A, B, C, D, and # were used as commands.

Table 5.5 shows the use for each key.

Table 5.5 Keys function.

Key character	Function	Description
A	Go. Starts the PWM	a) Convert setdigA/B/C on setpoint values (0-100%) b) Turn PWM on (those allowed) c) Clear counter
B	Stop. Stops the PWM	a) Decrement all duty cycles at the same time b) Turn off the PWM
C	Clear. Clear the setpoint	a) Clear dig1/2/3 and the counter b) Leaves same setpoints c) Reflects changes on the LCD's
D	Mode. Control selection	a) Toggle B4 of the m_word (0 power/1 current) b) Clear all setdig1/2/3 c) Reflects changes on the LCD's
#	Module selection. Controller module selection	a) Increment the content of module_w, changing from 1->2->3 and again.

Most of the commands call a subroutine to display the change on the LCD's.

Figure 5.34 shows the LCD's for the power and current mode.

```

Module  Dty %00 000 A
Setpoint: 00.0 kW Power 00.0 kW

```

a) Power mode

```

Module  Dty %00 000 A
Setpoint: 000 A Current 00.0 kW

```

b) Current mode

Figure 5.35. Display view: (a) Power mode and (b) Current mode.

The display of constants values was done using a subroutine inserted into the keyboard scan. The display of the current, voltage, duty cycle from the controller modules and the error signals was done in sequence. On the first loop, the value of the current was displayed, on the second the voltage and so on. This was done to avoid excessive time in the display of all the values. The limit to display each character on a fixed time it may be a problem if numerous characters has to be displayed.

The control for the modules depends on two different subroutines, one to scan the values of the ATD port and the other to control the modules based on the setpoint and the actual value, either power or current.

The setpoint of the user was converted onto a normalized number o 0 to 1, or a number for 0 to 65535, independently of the control mode (power or current). The control mode then calculated the power of the module based on two parameters, the output current and the duty cycle of the controller module. As long as the input voltage is constant (~630V) and the output voltage depends on the duty cycle, we can calculate the instantaneous voltage output. By multiplying the output voltage times the current, we obtain the power. The power has to be normalized also on fractional values. For the current applies the same. The output current is taken directly and then normalized.

Once we have the power and the current, we call the control routine. This concept is the same as applied for the increment-decrement control of the last approach, but now the PWM output is converted into a Dc signal that is used as the setpoint for the controller module. The frequency of the control is 1kHz, so the reference is almost constant. The control is before. The setpoint is compared against the reference of the user (power or current) and according to the actual value the duty cyclke in incremented or decremented to adjust the values.

It can be considered that this control can be risky if we consider there are two feedbacks for the same purpose.

CHAPTER 6

CONCLUSIONS

The objective of the project was to control up to 100kWatts. Practical test on field shows that it is possible to melt the soil with more that 50kWatts.

The first design implemented to control the power converters present problems. The low resolution of the ATD converter on the 68HC12, the resolution on the PWM, the configuration of the IGBTs and the inductance and the low capacitors at the input causes instability on the control using the 68HC12. One option could be the use of the fuzzy control module on the 68HC12.

The second design was more stable for different reasons. The control with the UC1823A was stable and very precise. To split the IGBTs, rectifier bridges and capacitances into separate modules proves to be a good option to protect the modules from each other.

The 68HC12 shows as a good option to interface the user with LCD and keyboard. Nonetheless the microcontroller was instable on the field due to the electric noise.

With more time, further improvements can be made to ensure a correct interface between the mutiplexer and the 68HC12. The simulations also show that it is recommended that the inductances at the output of the modules are equal and also the setpoints of the modules.

REFERENCES

1. Mohan, Ned. Power Electronics. John Wiley & Sons, New York, 1995.
2. Bayne, B. Stephen. High Power Semiconductor Switching and Reliability, Thesis, Texas Tech University, Dec.1994.
3. Benda, Vitezslav. Power Semiconductor Devices. John Wiley & Sons, New York, 1994.
4. Blicher, Adolph. Thyristor Physics. Springer-Verlag, New York, 1976.
5. Cogdell, J.R. Foundations of Electrical Engineering. Prentice Hall, Inc., Englewood Cliffs, NJ, 1996.
6. Laster, Clay. Thyristor Theory and Application. Tab Books Inc., Blue Ridge Summit, PA, 1986.
7. Portnoy, W.M. "Thyristors as Switching Elements for Pulsed Power," Pulsed Power Lecture Series, Lecture no. 23. 1980.
8. Streetman, Ben G. Solid State Electronic Devices. Prentice Hall Series in Solid State Physical Electronics, Upper Saddle River, New Jersey, 2000.
9. Motorola Semiconductor. Technical Data MC68HC912B32. Motorola INC. 1997
10. Spasov, Peter. Microcontroller Technology. Regent/Prentice Hall. Englewood Cliffs, NJ. 1993
11. Motorola. Evaluation Board, M68EVB912B32 User's Manual. Motorola INC. 1997
12. Motorola. 68HC12 Reference Manual. Motorola INC. 1997
13. Lipovski, Jack G. Microcontroller interfacing. Academic Press. San Diego, CA. 1999

APPENDIX A
68HC12 CODE

```

;-----
;   Average routine
;
;   For the three routines is
;   ( pointer2 - pointer1 ) / gainh
;-----
;   Sensor 1
;   clr      sumh          ;
;   clra          ;
;   clrb          ; Table from 900h to 91F (32d)
;   ldx      #$093e        ; (64d)      pointer1
;   ldy      #$093f        ;           pointer2
avg1: movb     1,X-,1,Y-    ;
;   movb     1,Y,suml      ; move the present value to
;   addd     sumh          ; the low byte
;   cpx      #$0900        ; Beggining of the table
;   bne      avg1          ;
;   movb     ADR4H,1,Y     ; save the ADC value in a temporary location
;   movb     1,Y,suml      ;
;   addd     sumh          ;
;   ldx      gainh         ; (127d)
;   idiv          ; (diff) / X
;   stx      current1h     ; Load the average
;-----
;   Sensor 2
;   clr      sumh          ;
;   clra          ;
;   clrb          ; Table from 940h to 95F (32d)
;   ldx      #$097e        ; Points begining of the average table
;   ldy      #$097f        ;
avg2: movb     1,X-,1,Y-    ;
;   movb     1,Y,suml      ; move the present value to
;   addd     sumh          ; the low byte
;   cpx      #$0940        ; Beggining of the table
;   bne      avg2          ;
;   movb     ADR5H,1,Y     ; save the ADC value in a temporary location
;   movb     1,Y,suml      ;
;   addd     sumh          ;
;   ldx      gainh         ;
;   idiv          ; (Sum_table) / X
;   stx      current2h     ; Load the average
;-----
;   Sensor 3
;   clr      sumh          ;
;   clra          ;
;   clrb          ; Table from 980h to 91F (32d)
;   ldx      #$09be        ; Points begining of the average table
;   ldy      #$09bf        ;
avg3: movb     1,X-,1,Y-    ;
;   movb     1,Y,suml      ; move the present value to
;   addd     sumh          ; the low byte
;   cpx      #$0980        ; Beggining of the table
;   bne      avg3          ;
;   movb     ADR6H,1,Y     ; save the ADC value in a temporary location
;   movb     1,Y,suml      ;
;   addd     sumh          ;
;   ldx      gainh         ;
;   idiv          ; (Sum_table) / X
;   stx      current3h     ; Load the average
;-----
;   Total current
;-----
;   ldd      curr1
;   addd     curr2
;   addd     curr3
;   std      currtoth
;-----

```

A.1 Average Routine code.

```

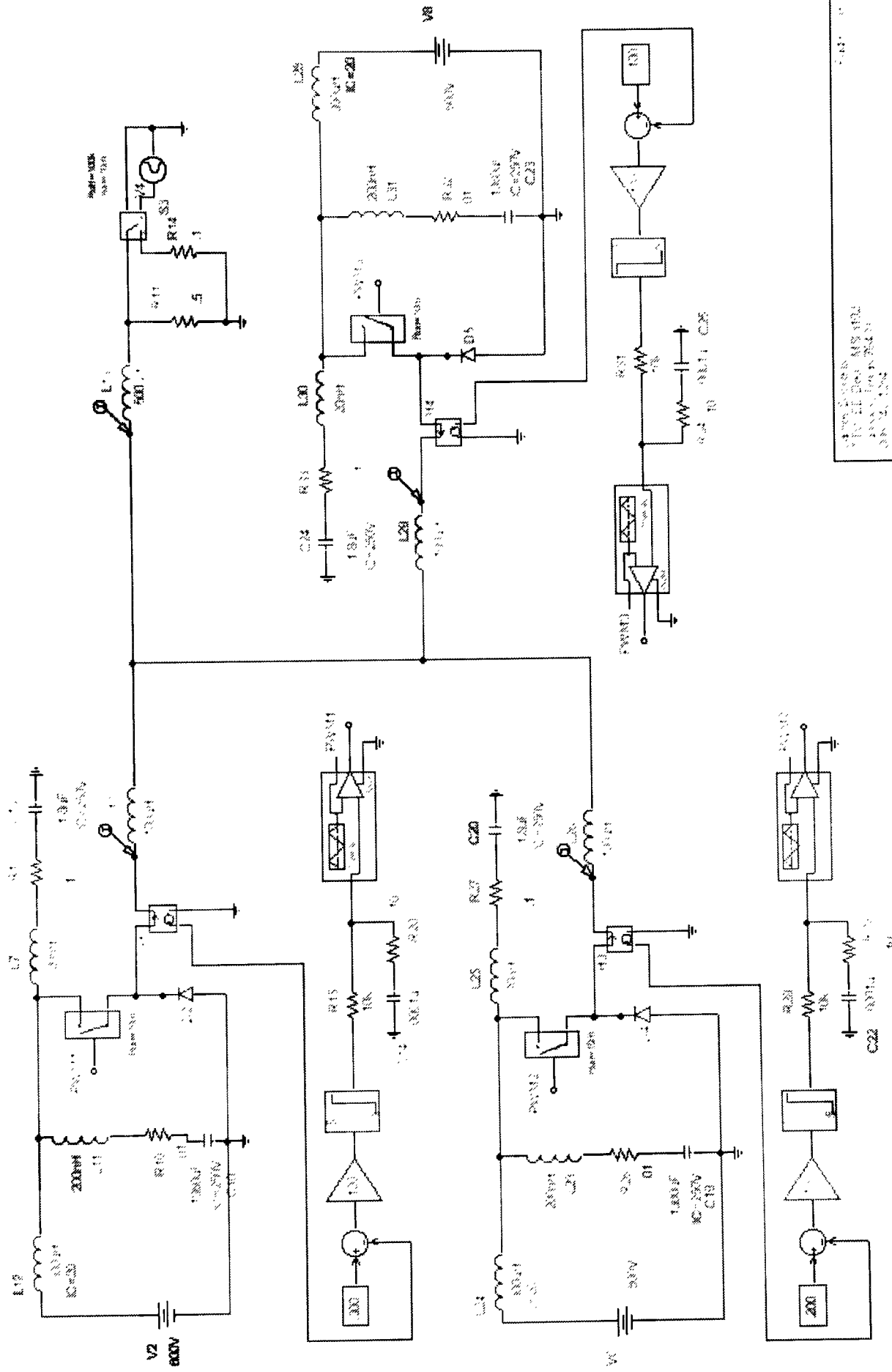
*****
*      MAIN PROGRAM
*****
loop:
    ldaa    ADR2H        ; A -> analogical input
    staa    set_point1   ; A = Setpoint
;-----;
;      proportional control
;
    ldab    ADR4H        ;
    stab    current1h     ; B = Current1
    sba     ; A - B -> A, error = Set-Curr
    bcs     end           ; if negative, go to end
    ldab    #$C8          ; to adjust the duty cicle
    mul     ; to C8h (10kHz)
    staa    error         ;
    exg     A,B           ;
    ldaa    #$00          ;
    ldy     #$0001        ; Y = Proportional Gain
    emul    ;
    cpd     #$00C7        ;
    bcc     end           ;
    stab    PWDTY0        ; if not,
    stab    PWDTY1        ;
end:
;-----;
;      Sampling frequency
;-----;
;      ldx     #$003F      ; Delay for a sample frequency of
;b1    dex     ; 1fh = 20kHz
;      bne     b1         ; 00h 80kHz
;-----;

```

A.2 Proportional control code.

APPENDIX B SIMULATIONS

Buck (Step-Down) Converter for Plasma Burner



+V2=600V
 +V8=500V
 +V1=0V
 +V2=600V
 +V8=500V

Figure B.1 Plasma burner.

Buck (Step-Down) Converter for Plasma Burner

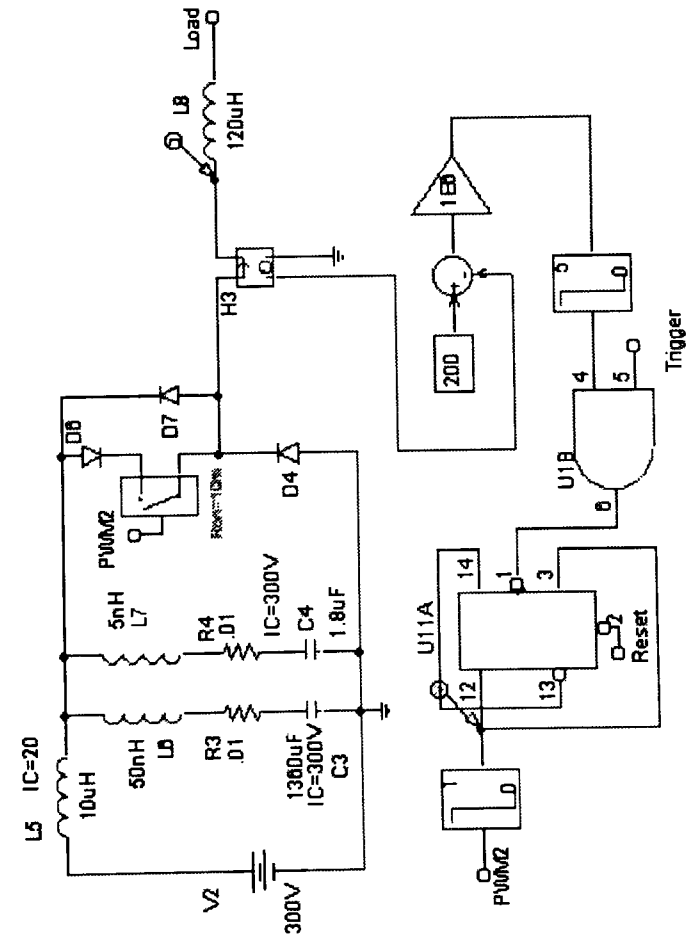
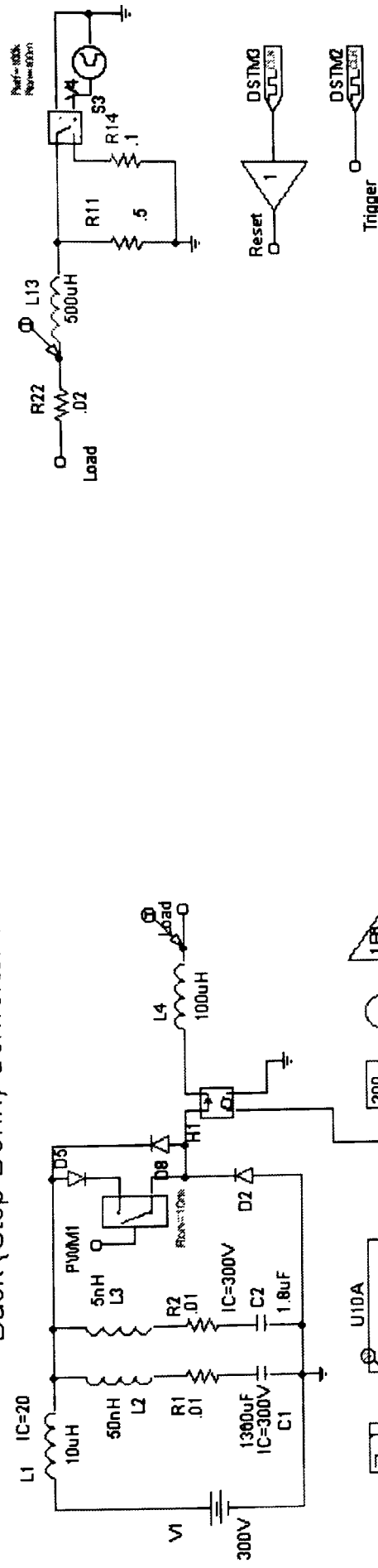


Figure B.2 Current mode controller.

APPENDIX C
CONTROLLER BOARDS

HC12 Interface for Semikron 1442 GAR 120 - 414 CTV

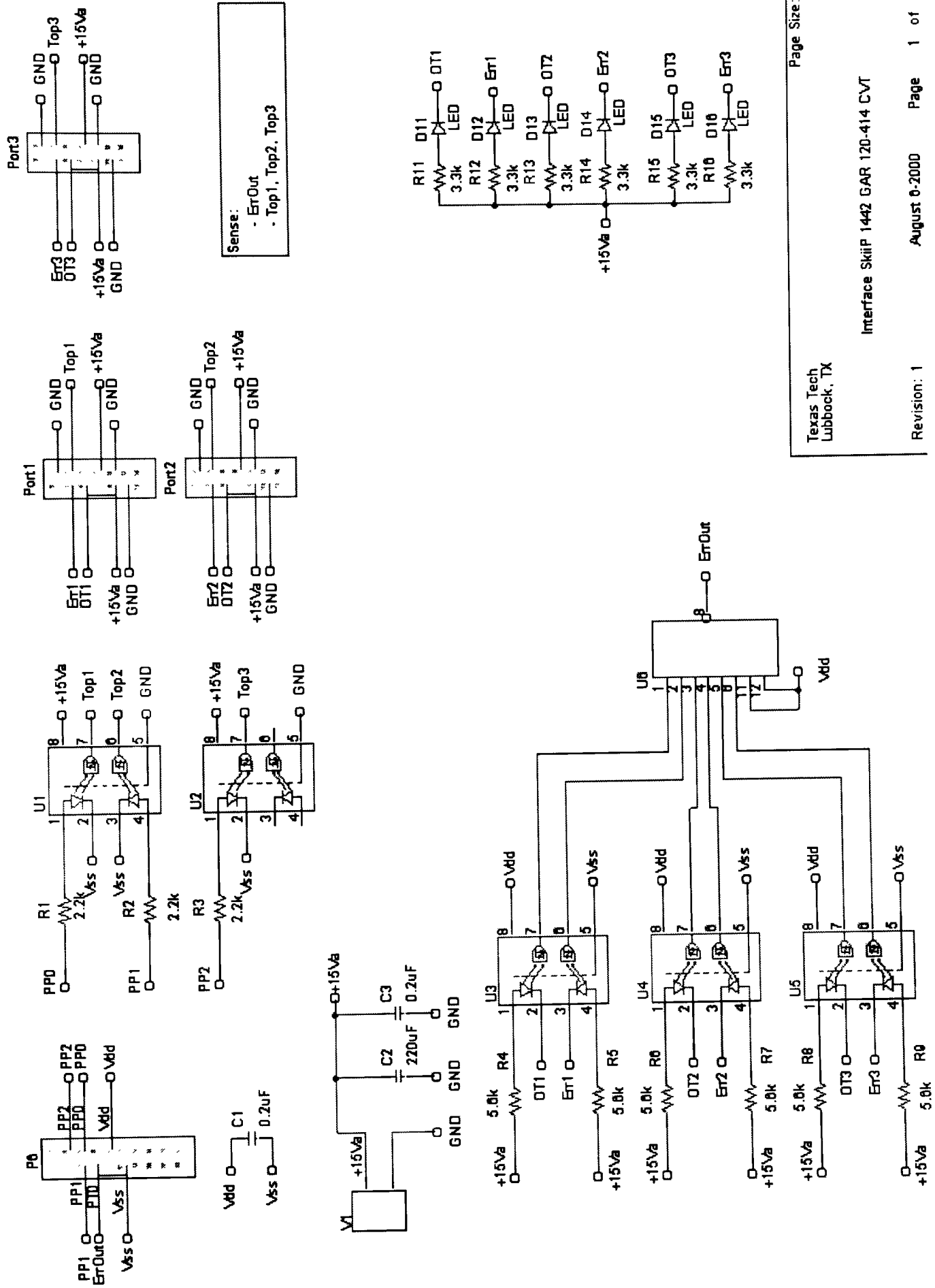


Figure C.1. 68HC12 Interface with the IGBT's.

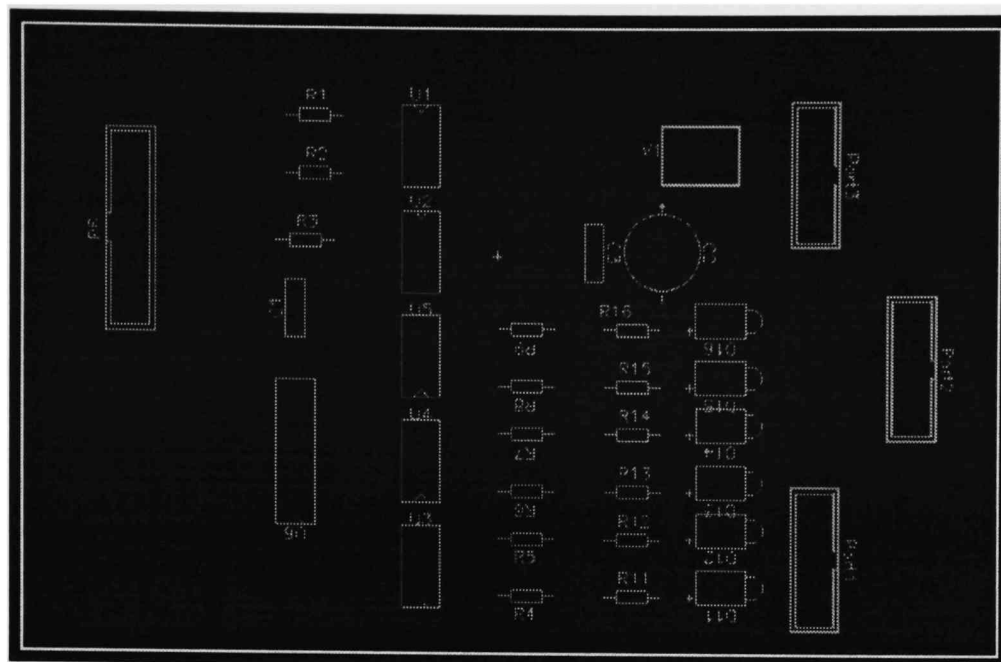


Figure C.2 Pcboard silk side of diagram on Figure C.1.

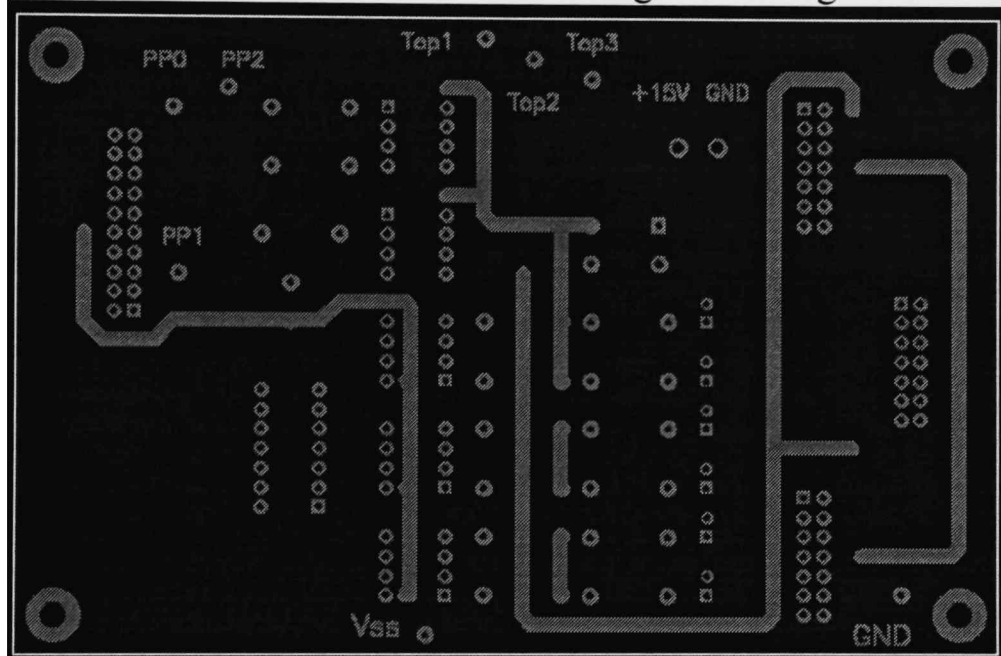


Figure C.3 Pcboard component side of diagram on Figure C.1.

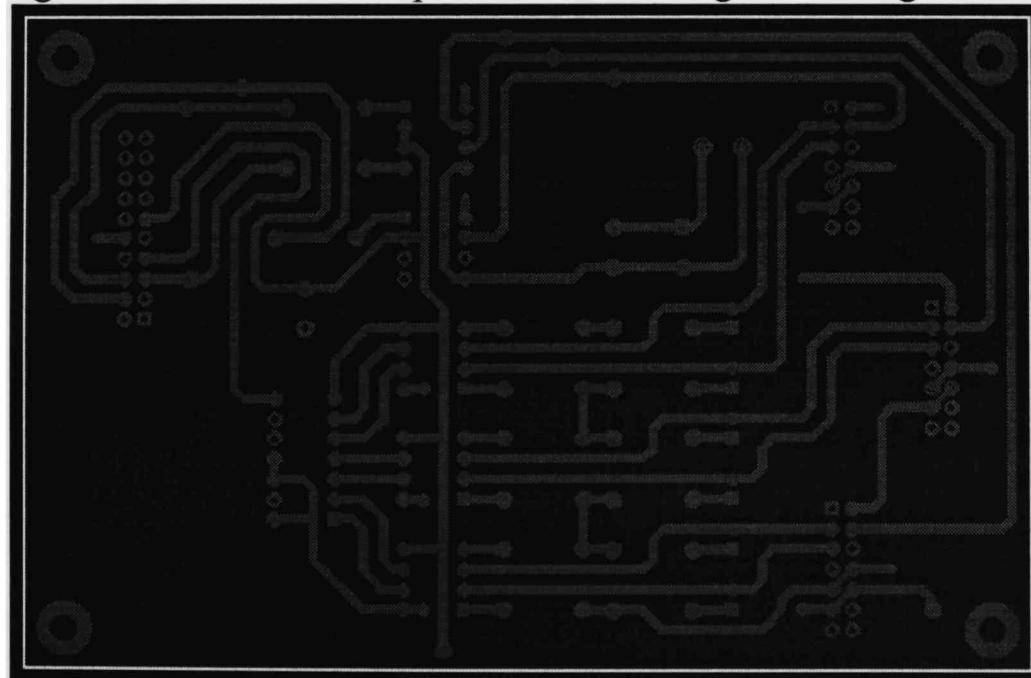


Figure C.4. Pcboard solder side of diagram on Figure C.1.

HC12 Sensors Interface for the Torch project

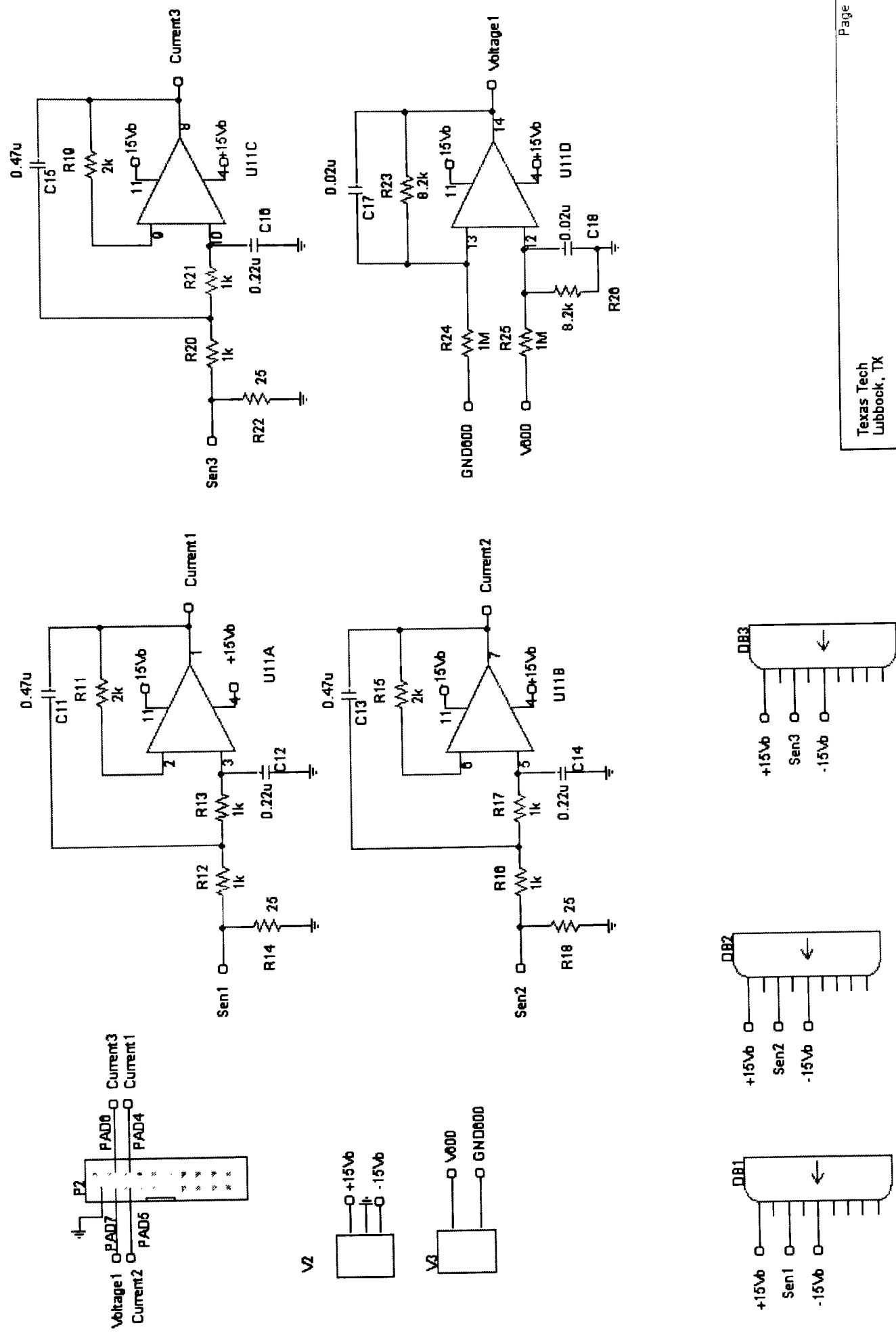


Figure C.5 Schematic of the Analog to digital interface.

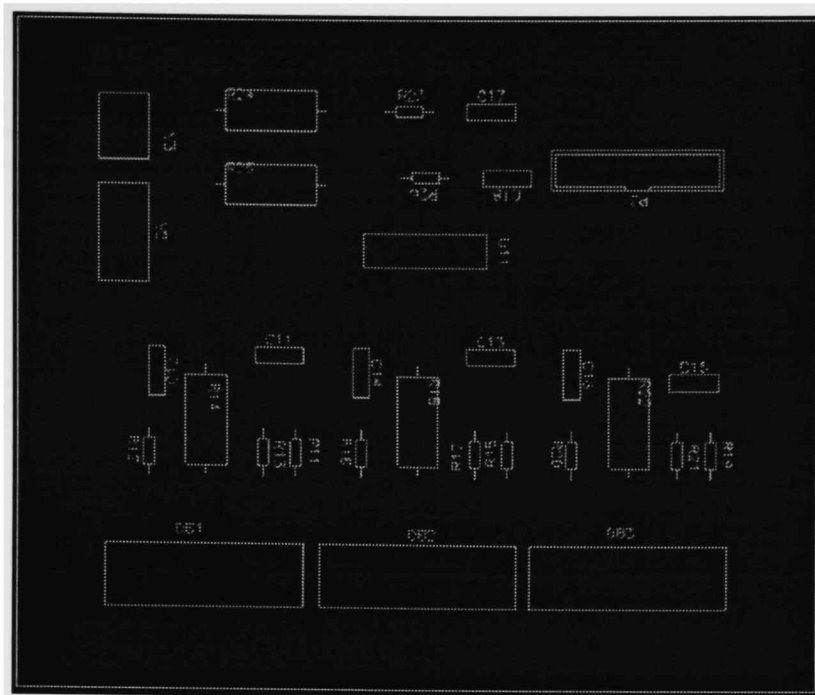


Figure C.6 Pcboard silk side of diagram on Figure C.5.

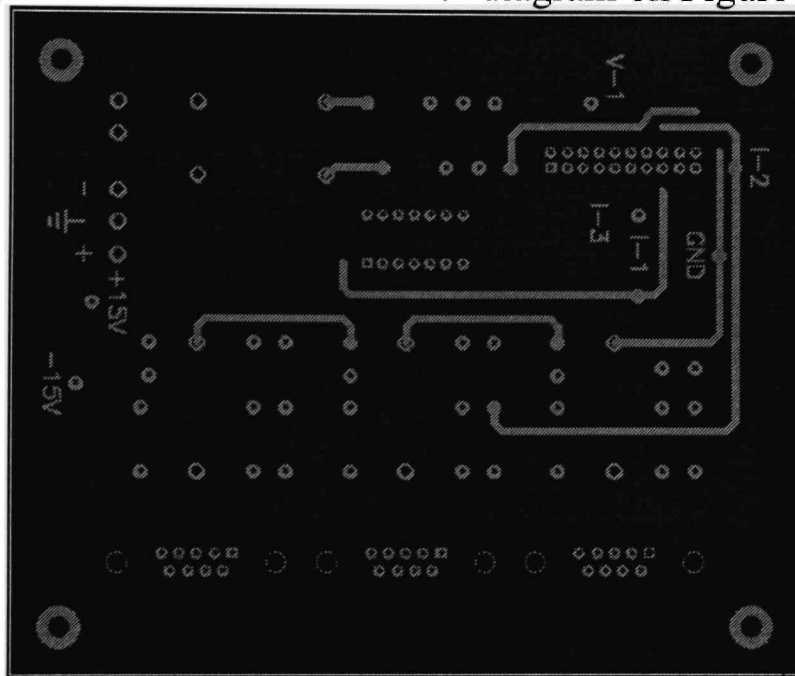


Figure C.7 Pcboard component side of diagram on Figure C.5.

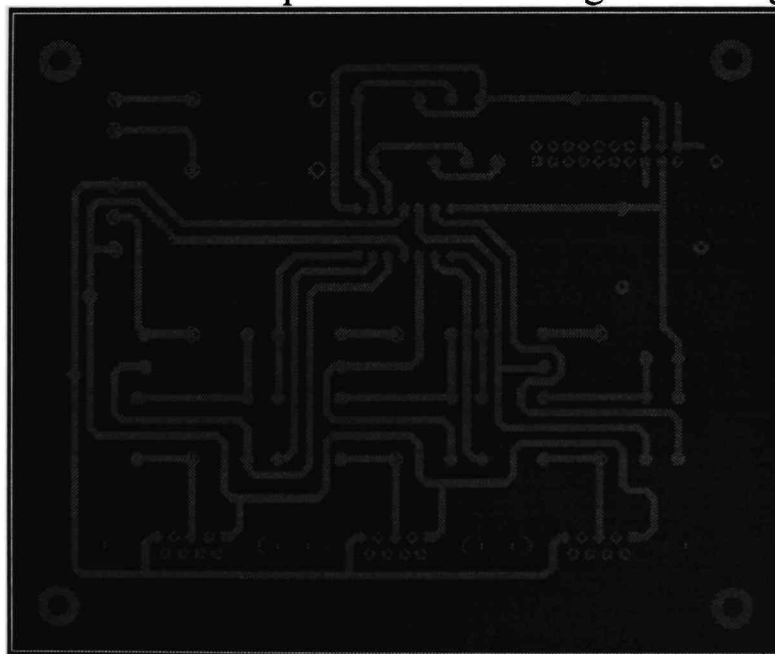


Figure C.8 Pcboard solder side of diagram on Figure C.5.

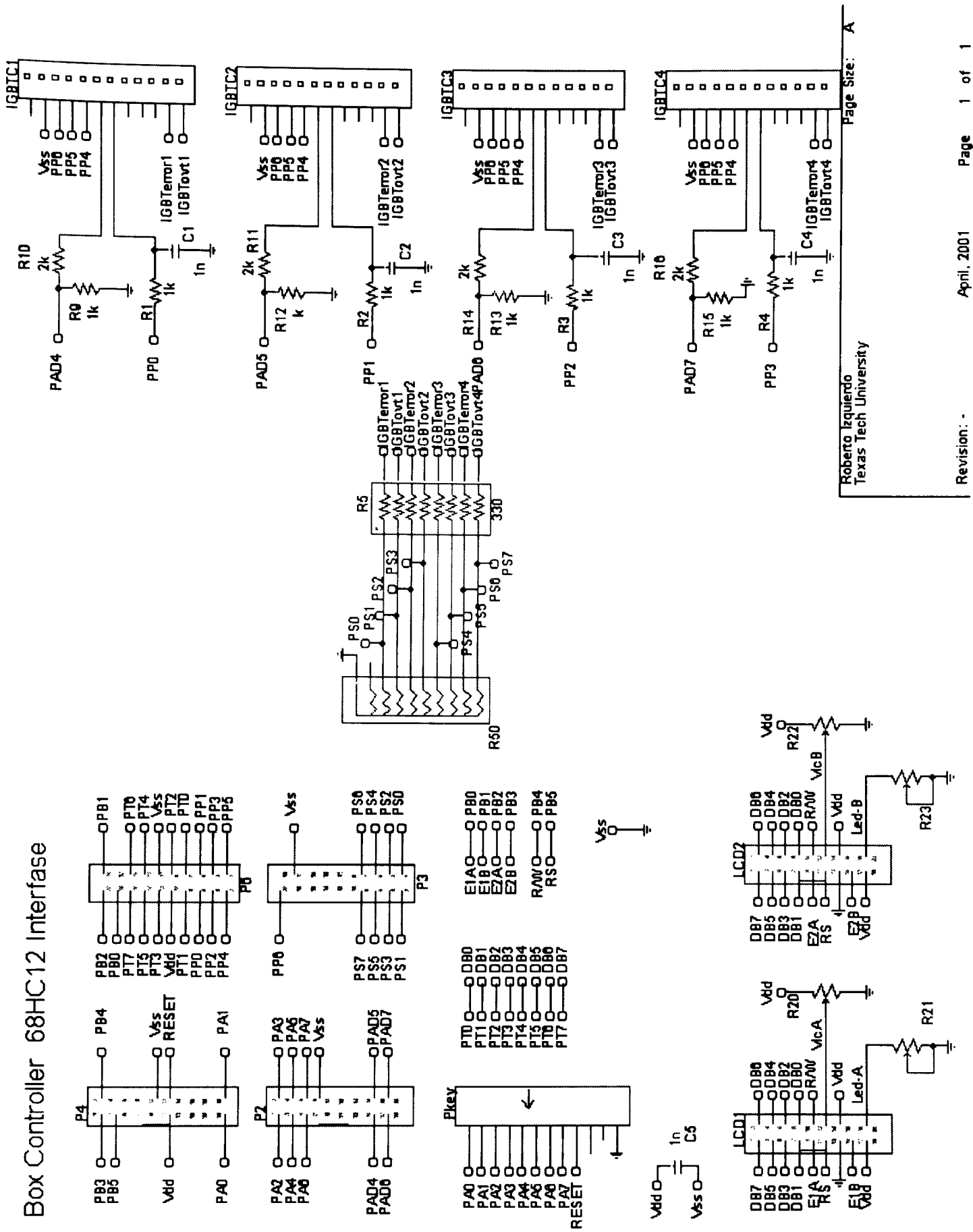


Figure C.9 Controller box interface between the HC12 and the keyboard-LCD

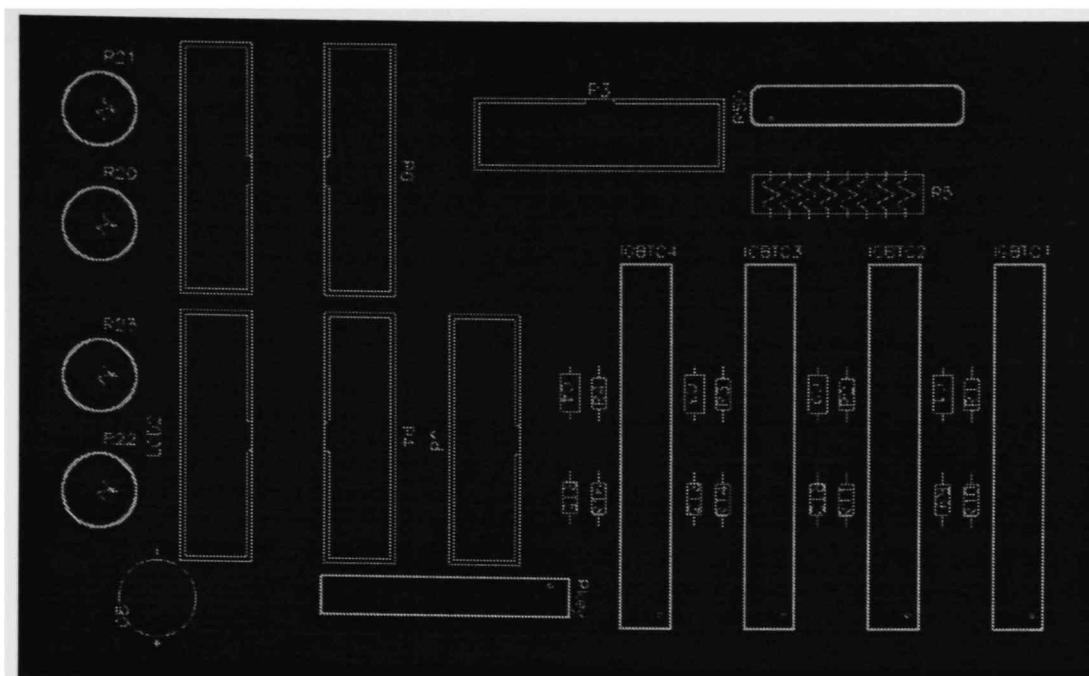


Figure C.10 Pcboard silk side of diagram on Figure C.9.

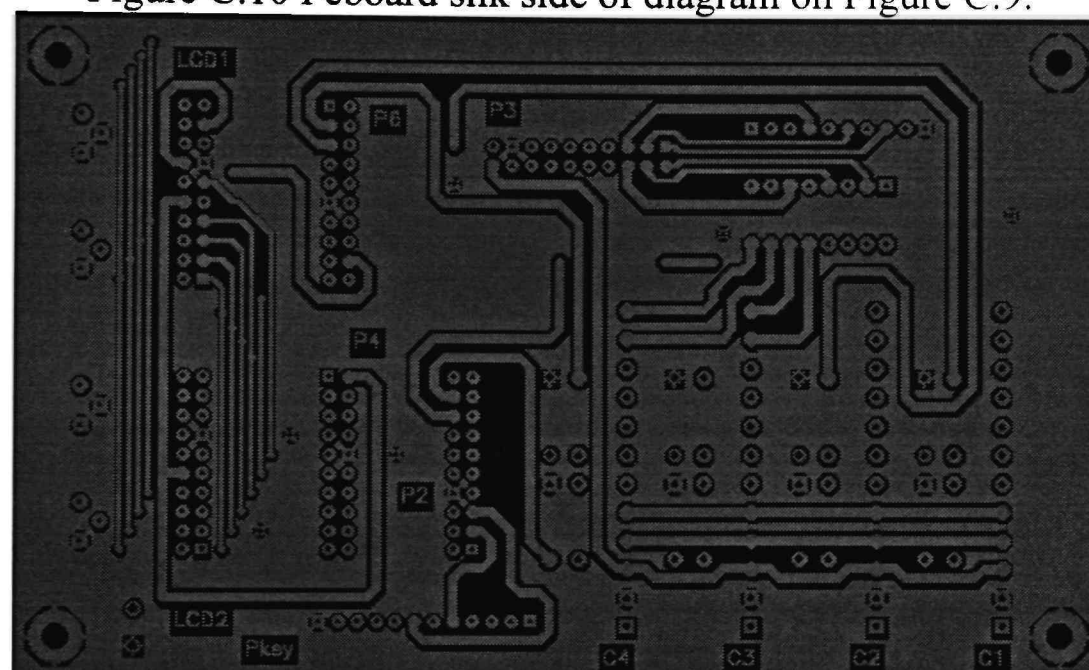


Figure C.11 Pcboard component side of diagram on Figure C.9.

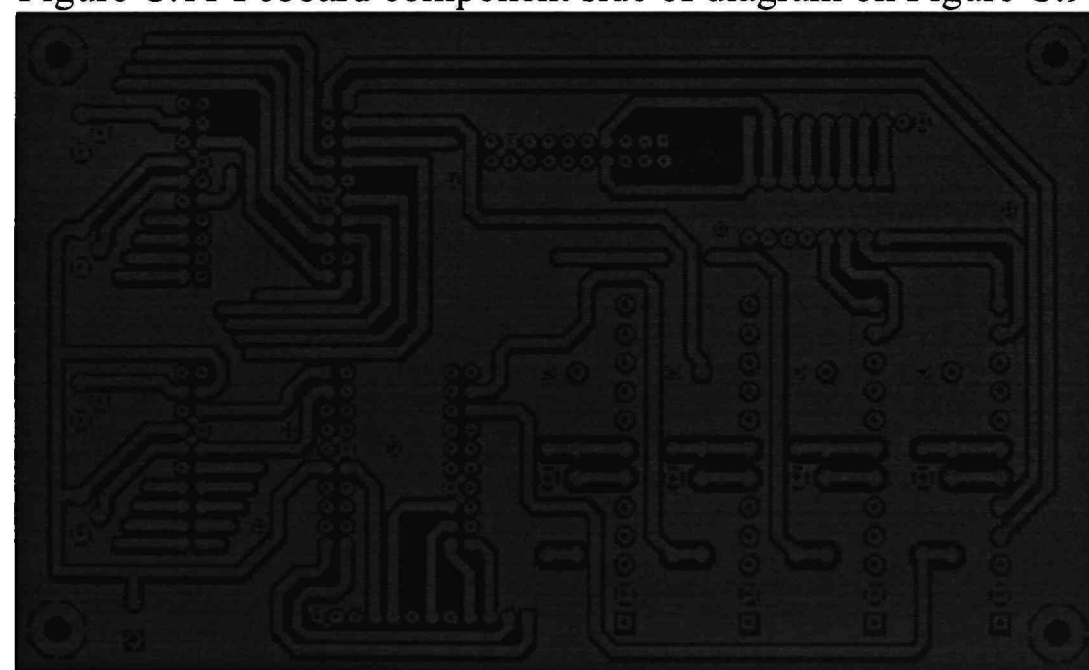


Figure C.12 Pcboard solder side of diagram on Figure C.9.

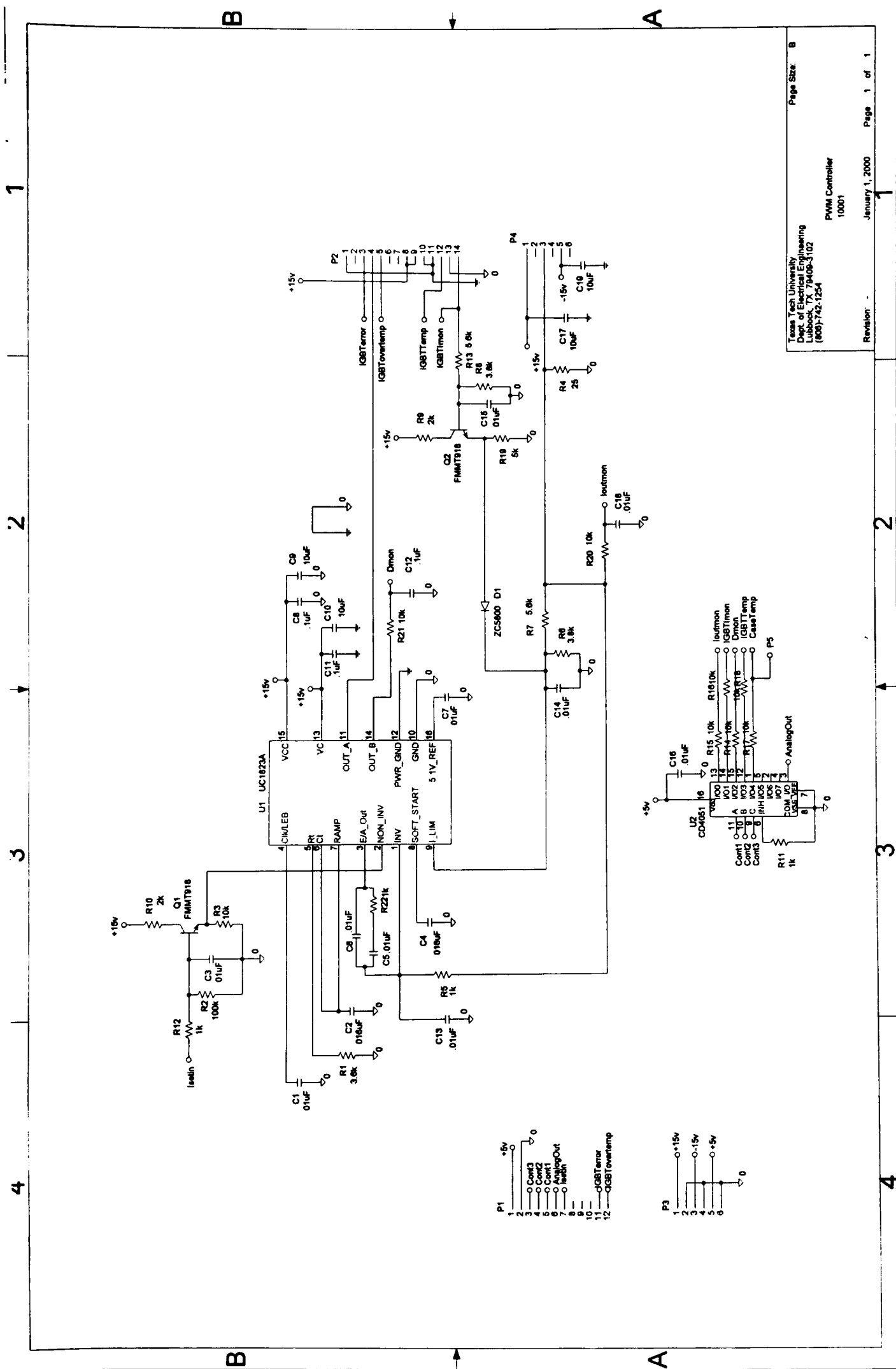


Figure C.13. PWM controller schematic.

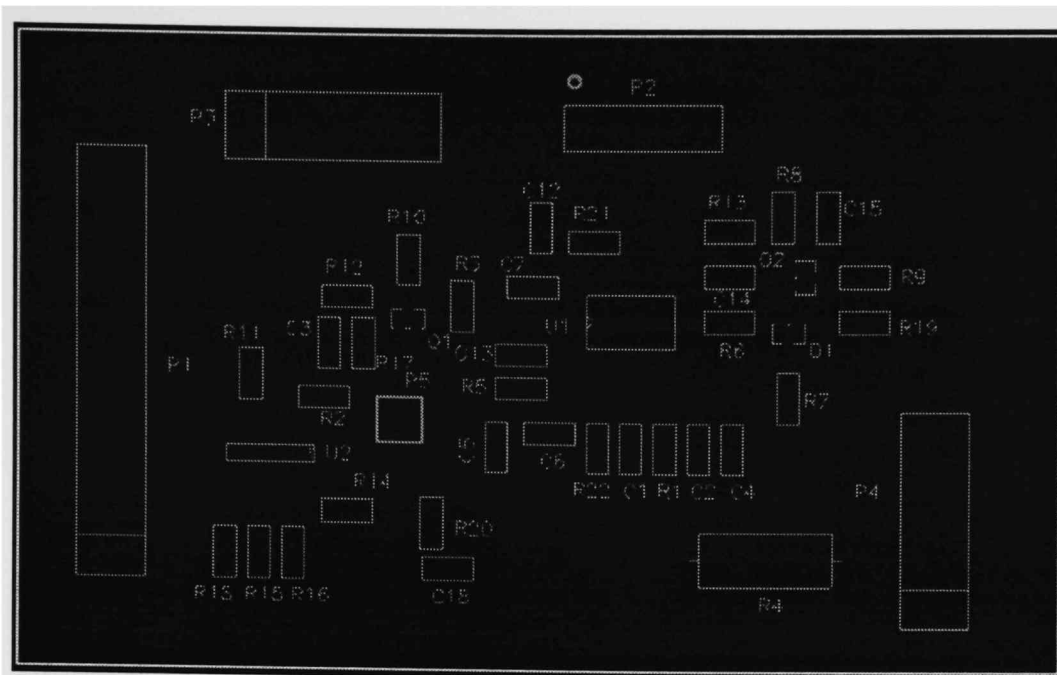


Figure C.14 Pcbord silk side of diagram on Figure C.13.

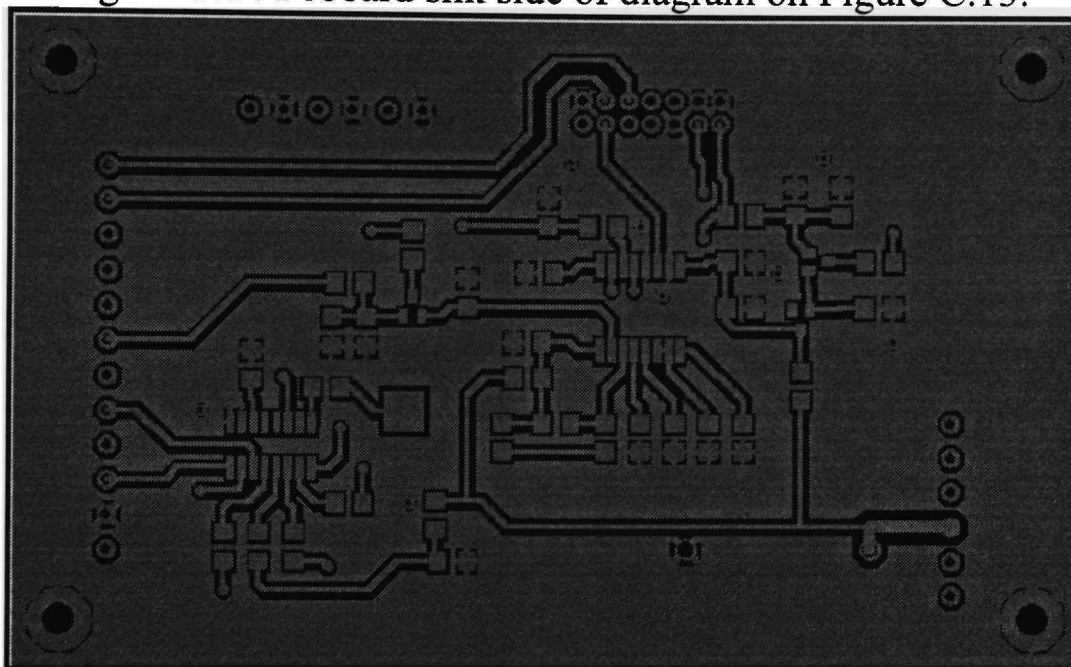


Figure C.15 Pcbord component side of diagram on Figure C.13.

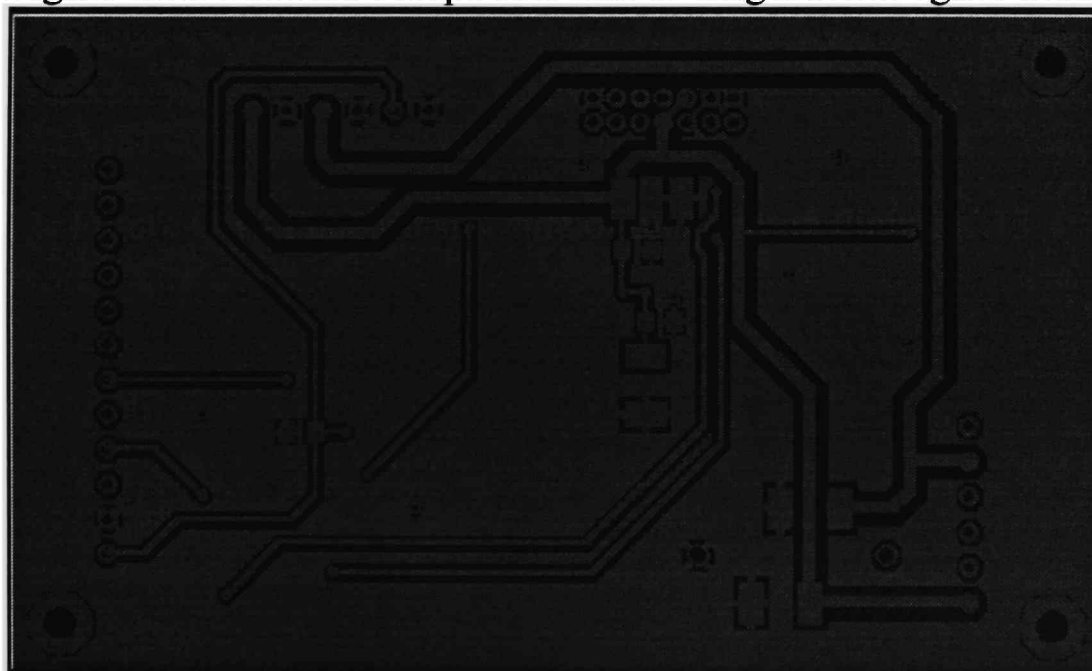


Figure C.16 Pcbord solder side of diagram on Figure C.13.

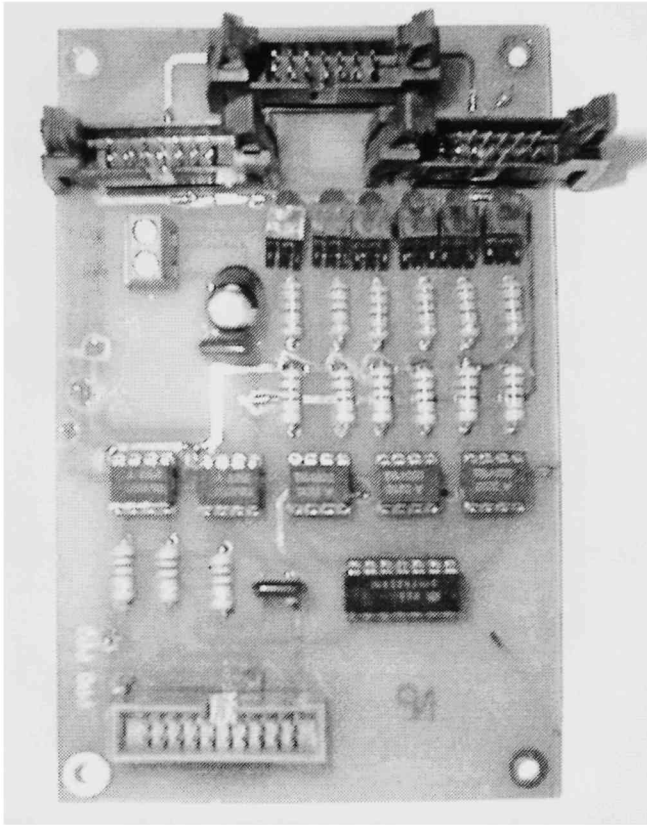


Figure C.17 Controller Interface

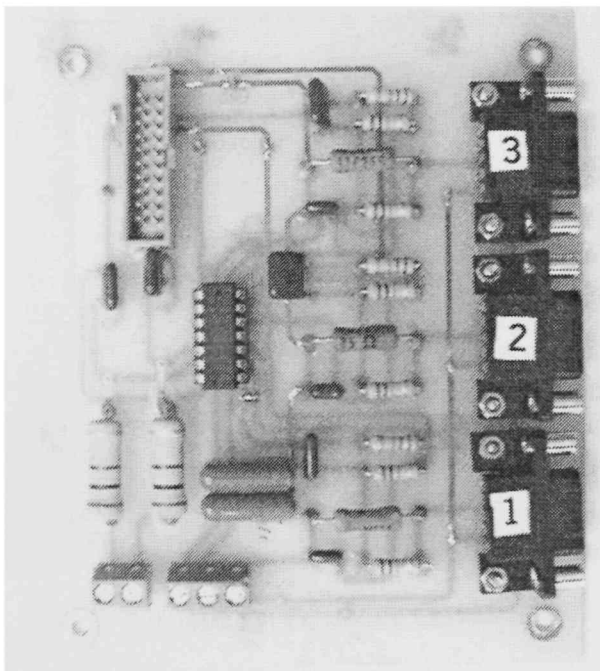


Figure C.18 Analog Interface

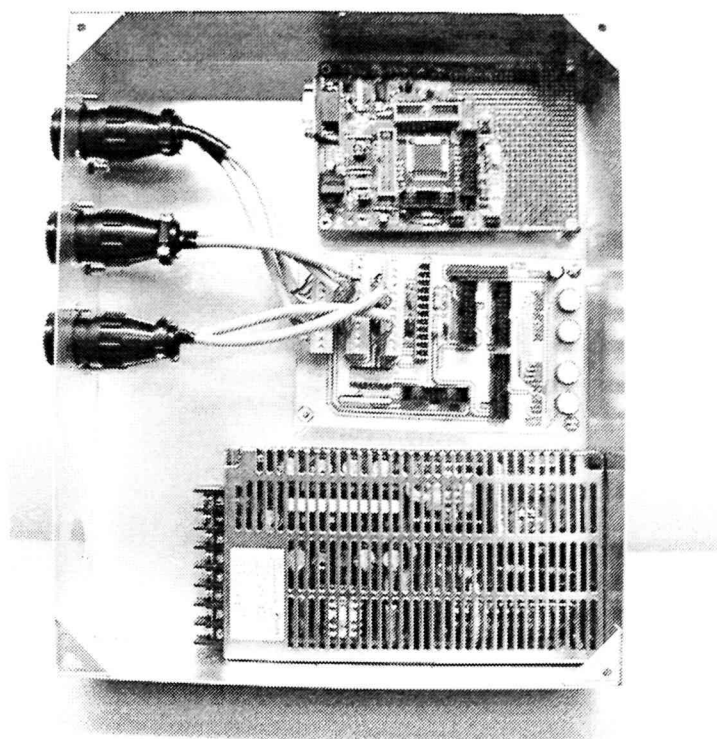


Figure C.19 68HC12 General Interface

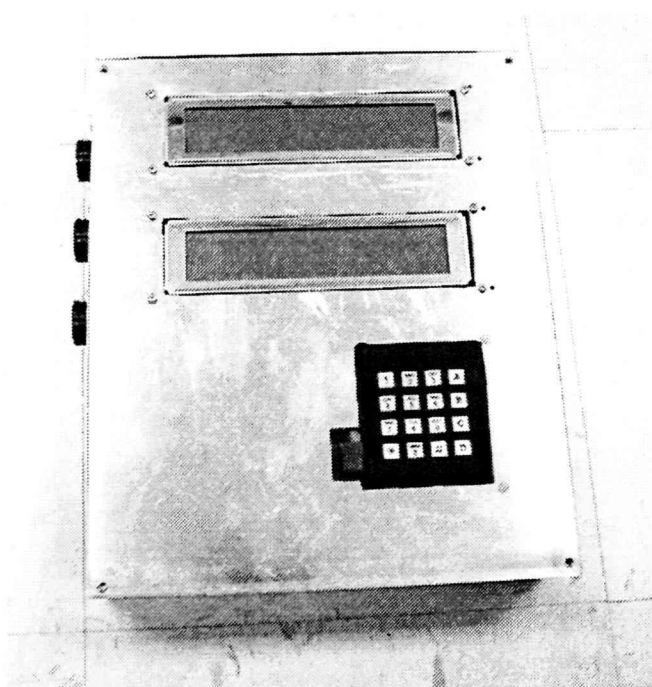


Figure C.20 Box cover.

APPENDIX D
TORCH TRUCK

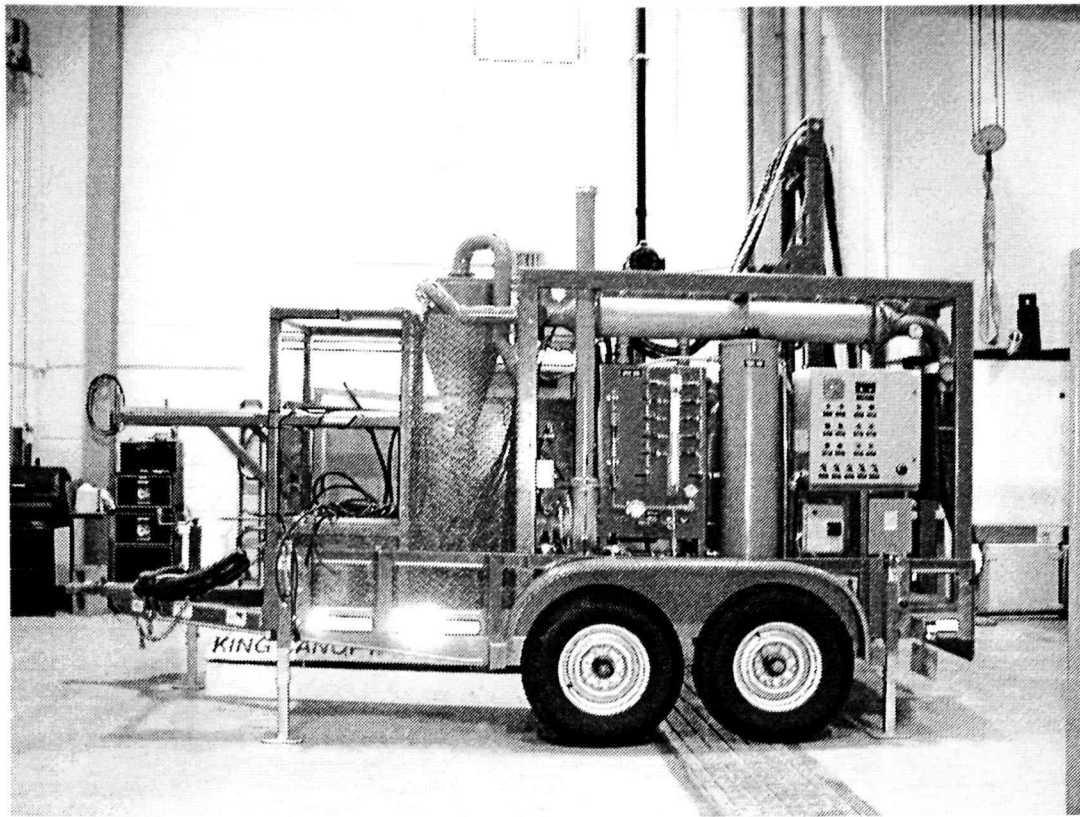


Figure D.1 Mechanical system

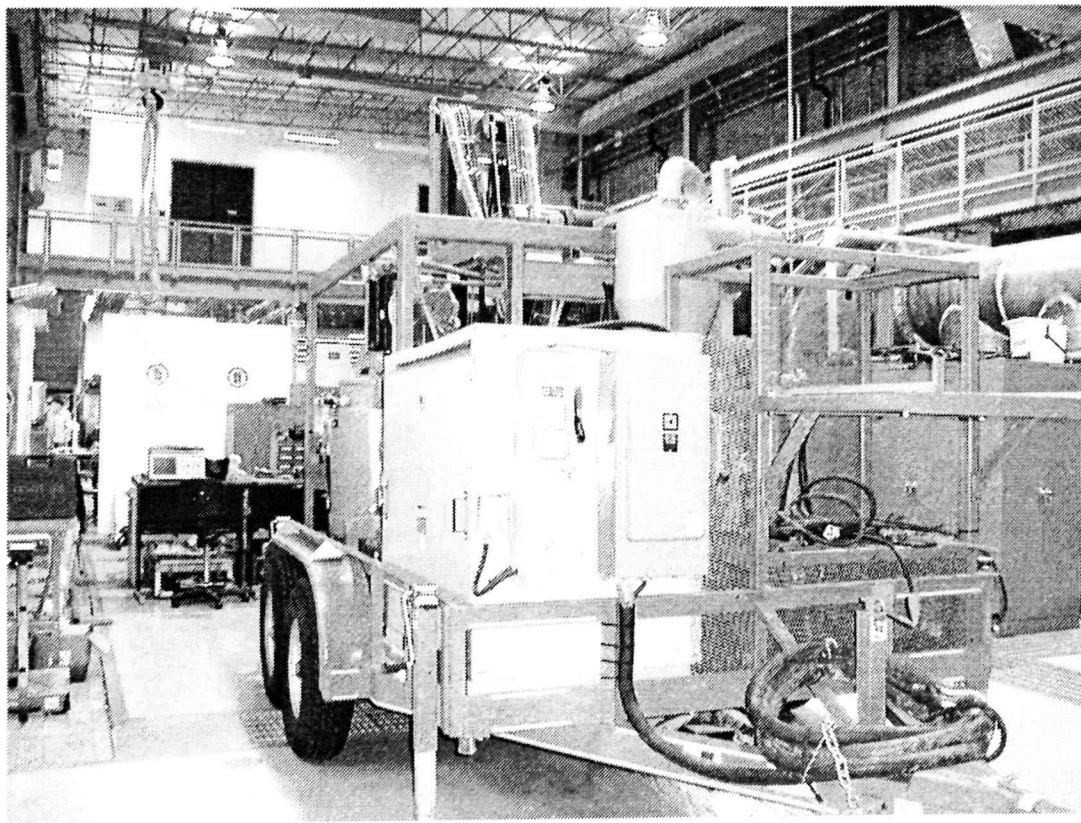


Figure D.2 Transformer.

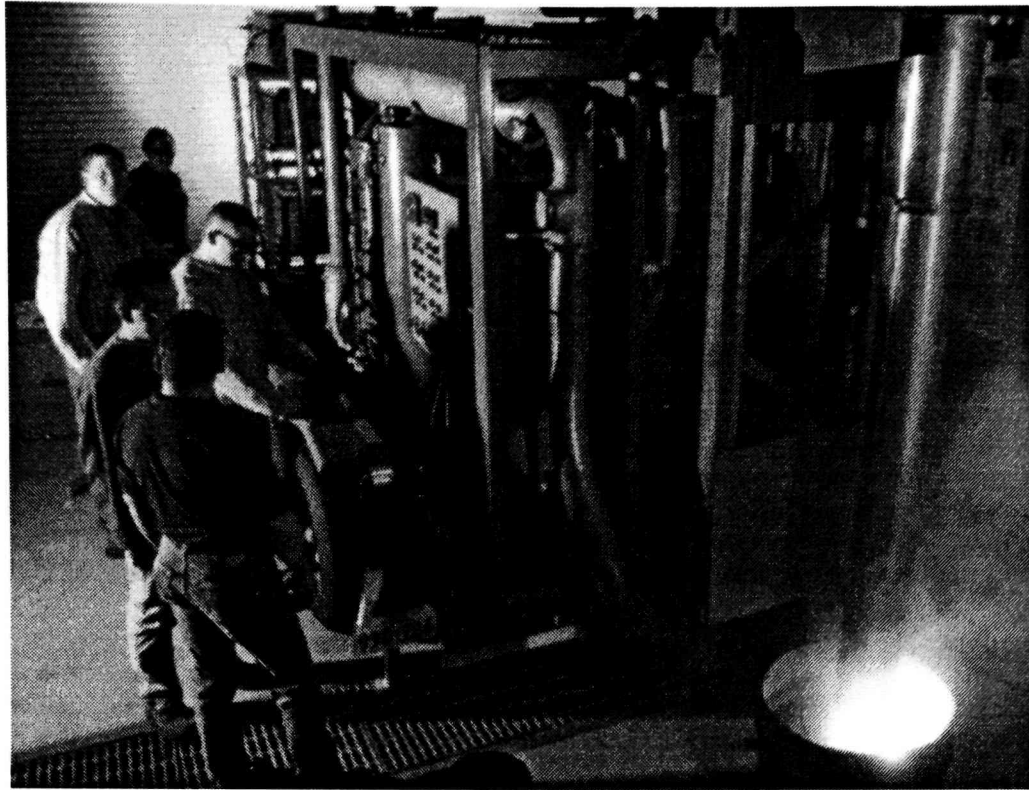


Figure D.3 Power converter operation.

PERMISSION TO COPY

In presenting this thesis in partial fulfillment of the requirements for a master's degree at Texas Tech University or Texas Tech University Health Sciences Center, I agree that the Library and my major department shall make it freely available for research purposes. Permission to copy this thesis for scholarly purposes may be granted by the Director of the Library or my major professor. It is understood that any copying or publication of this thesis for financial gain shall not be allowed without my further written permission and that any user may be liable for copyright infringement.

Agree (Permission is granted.)

Student Signature

Date

Disagree (Permission is not granted.)

Student Signature

Date